UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

TRABAJO DE FIN DE GRADO

GRADO EN INGENIERÍA DE SISTEMAS DE COMUNICACIONES

SISTEMA PARA EL ALINEAMIENTO DE SUBTÍTULOS Y AUDIO EN ESCENARIOS DE REHABLADO EN TV

AUTOR: ALEJANDRO LOZANO TORRIJOS

TUTORA: MERCEDES DE CASTRO ÁLVAREZ

DIRECTOR: DIEGO CARRERO FIGUEROA

JUNIO DE 2012

Prólogo

A lo largo de toda la historia la información ha sido un elemento crucial para la vida de las personas. Ansiada por muchos y utilizada por otros, la información ha supuesto un elemento de vital importancia desde tiempos inmemoriales. Una constante siempre presente durante la evolución de la humanidad hacia las sociedades actuales es el progreso en materia de telecomunicaciones. Es básico que las personas puedan transmitir sus conocimientos y descubrimientos, y para ello se idearon cada vez más novedosas formas de comunicar. Con este progreso, también se ha ido logrando a lo largo de los años que cada vez menos personas sean incapaces de acceder a la información. De hecho, actualmente nos encontramos en una sociedad de la información, regida por las tecnologías de la información y comunicación (las TIC).

Una segunda constante siempre ha sido la existencia de determinados sectores de la sociedad incapaces de acceder a la información. Antaño por imposibilidad tecnológica, pero por fortuna dicho problema está solventado en muchos lugares. A pesar de esto aún hay una barrera muy importante que perdura: no todas las personas son capaces de utilizar los medios habituales para acceder a la información. Hoy día cualquiera puede ver la televisión, leer un periódico o escuchar la radio para conocer la situación del mundo que les rodea, pero eso no es posible para algunos sujetos de la sociedad, que por desgracia padecen de algún tipo de discapacidad. Entre ellas, las que más impacto tienen en la capacidad de comunicación son la discapacidad visual y la auditiva.

Esta situación es dramática, y por ello es necesario desarrollar herramientas que permitan a esos individuos hacer uso de esos medios de comunicación. En otras palabras, utilizando la tecnología es posible mejorar la calidad de vida de esas personas, brindándoles la oportunidad de valerse por sí mismos y de poder ver un programa de televisión pese a su discapacidad auditiva, por ejemplo, o de poder entender una obra de teatro pese a su incapacidad visual. En este sentido, entidades como las cadenas de televisión llevan años tomando medidas para ofrecer porcentajes mínimos de programación subtitulada o audiodescrita, pero aún quedan aspectos tecnológicos de crucial importancia para los usuarios que están pendientes de resolver.

A lo largo de la presente memoria se describe el diseño, desarrollo y funcionamiento del proyecto de sincronización de subtítulos en escenarios de rehablado en televisión. Éste consiste en la implementación de una herramienta capaz de obtener subtítulos bien formados y sincronizados con el audio y el vídeo de una emisión determinada. Esta herramienta podría permitir solventar o, al menos, atenuar los graves problemas de sincronismo a la hora de subtitular programas de televisión en directo, escenario donde los retardos de lo plasmado en pantalla respecto al audio son el principal motivo de descontento entre las personas sordas.

Índice

Índice de figuras

Índice de tablas

1. Introducción

1.1 Motivación

Dentro del Centro Español de Subtitulado y Audiodescripción, grupo de investigación dependiente de la Universidad Carlos III de Madrid, se propuso el desarrollo de una solución que posibilitara paliar los problemas de sincronismo inherentes a la generación de subtítulos mediante técnicas de rehablado para programas de televisión en directo.

Los más que notables retardos que aparecen en estos escenarios no sólo son incómodos y molestos, sino que en una gran cantidad de ocasiones impiden al televidente comprender lo que está sucediendo. Es más que frecuente comprobar que muchas veces los subtítulos que se presentan en pantalla en un instante dado son acerca de algo que se dijo con varios segundos de anterioridad. Esto, que puede parecer un problema sin importancia, es algo muy grave en según qué casos para las personas con discapacidad auditiva. Por ejemplo, en un debate político un error de estas características es desastroso, puesto que aparece en pantalla uno de los contendientes hablando y el subtitulado se refiere a lo que dijo su adversario unos segundos antes. En otras ocasiones, durante la emisión de una noticia, el subtitulado se refiere a la noticia anterior[1]. Y como estos muchos otros ejemplos.

Es evidente pues que se necesita una solución a estos problemas. La coherencia temporal entre lo que se muestra en pantalla y el texto que se expone en la misma es crucial, y eso es lo que se pretende obtener con el presente proyecto. Así mismo, se busca una generación de subtítulos que se ajuste en la medida de lo posible a los estándares existentes de subtitulado. Otro motivo además del moral por el que se requiere una solución a los graves problemas existentes en esta materia es que una persona discapacitada debe poder acceder a la información, por ley. Y en caso de que unos subtítulos tardíos o mal formados lo impidiesen, se estaría incumpliendo la Ley de igualdad de oportunidades, no discriminación y accesibilidad universal de las personas con discapacidad, Ley 51/2003.

Actualmente no hay ninguna solución real que se esté utilizando para solucionar la situación descrita en los párrafos anteriores. Por ello, cualquier mejora en cuanto a la misma será gratamente acogida por el sector de la sociedad que la sufre. Toda mejora supone un cambio muy favorable, sobre todo en un ámbito en el que la tecnología aún tiene mucho que evolucionar y en el que la única elección es entre disponer de unos subtítulos de muy baja calidad o no disponer de ellos.

[1] Se puede comprobar cualquiera de estas situaciones viendo la televisión y activando la opción de subtitulado en programas emitidos en directo. No obstante, se proporcionan algunos vídeos ilustrativos para que el lector pueda comprender la problemática:

1. Programa 59 segundos: http://www.youtube.com/watch?v=CnyMFtrizks&feature=youtu.be
2. Programa 59 segundos: http://www.youtube.com/watch?v=6r4kONLmfU0&feature=youtu.be
3. Noticias de la BBC: http://www.youtube.com/watch?v=kwJFUPwGRSo&feature=youtu.be

1.2 Contexto

El trabajo desarrollado se ha realizado dentro del Centro Español de Subtitulado y Audiodescripción (CESyA), en concreto dentro del marco del Instituto de Desarrollo Tecnológico y Promoción de la Innovación "Pedro Juan de Lastanosa", perteneciente a la Universidad Carlos III de Madrid. Asimismo, el CESyA es un centro dependiente del Real Patronato sobre Discapacidad – Ministerio de Sanidad, Servicios Sociales e Igualdad, cuyo proyecto multidisciplinar es favorecer la accesibilidad en los medios audiovisuales, a través de servicios de subtitulado y audiodescripción principalmente. Los objetivos de este grupo son aportar soluciones tecnológicas a los diferentes problemas que pudieran tener las personas por motivos de discapacidad. De igual manera, se pueden destacar entre sus objetivos la contribución en iniciativas de normalización, comunicación y sensibilización social sobre la accesibilidad audiovisual, así como la coordinación de acciones de formación e investigación homologada y la creación de una base de datos que contenga referencias al material subtitulado y audiodescrito disponible.

El proyecto que se describe en este documento supone la continuación de una línea de trabajo en la que el CESyA lleva investigando tres años. Este interés es debido a que esta problemática es algo pendiente de resolver en todas las televisiones del mundo que generen subtítulos en tiempo real, por lo que cualquier avance en este ámbito es absolutamente necesario.

1.3 Objetivos del proyecto

El presente proyecto tiene una serie de objetivos bien definidos. El principal y más importante es el de diseñar e implementar una herramienta capaz de solventar el principal problema derivado de subtitular programas de televisión en directo, que es la mala sincronización. Dentro de este ámbito hay una gran variedad de situaciones y entornos por lo que es necesario acotar el área de investigación para después, si los resultados son los esperados, profundizar en escenarios más complejos. En este caso concreto se pretende resolver el problema en escenarios de rehablado en televisión, particularmente en situaciones controladas, con audio limpio y las mínimas perturbaciones posibles.

Por otro lado, se pretende obtener un sistema capaz de subtitular de una manera relativamente mecanizada, esto es, que genere subtítulos bien sincronizados y formados de acuerdo a los estándares establecidos utilizando la menor cantidad de recursos humanos posible en entornos de directo o diferido. En los primeros porque sólo automatizando la tarea es posible mejorar los retardos existentes. En los segundos porque facilita enormemente la labor de subtitulado.

Por último, se trata de realizar un ejercicio de concienciación. Entender que los problemas expuestos existen y no están resueltos. En este caso, un ingeniero busca mejorar la vida de los demás buscando soluciones a dichos problemas valiéndose de la tecnología. Mejorar la calidad de vida de esos individuos es posible, pero muy

complicado y costoso. No son pocos los esfuerzos que se han realizado y se realizan en materia de accesibilidad, pero siempre es adecuado indagar y buscar nuevas soluciones, nuevos aportes que añadir a todo el trabajo existente que sean de utilidad para la sociedad.

1.4 Estructura del documento

La presente memoria presenta los diferentes aspectos que conforman el proyecto por completo. Este documento se compone de las siguientes secciones:

El capítulo 2 muestra una somera descripción del Estado del Arte en materia de subtitulado (y, más concretamente, de subtitulado en televisión), esto es, en qué consiste esta práctica, la casuística asociada y el marco regulatorio sobre el que se sustenta. Así mismo, se comenta sucintamente qué son los reconocedores de voz y cómo funcionan, para concluir con una breve descripción de algunos trabajos relacionados en este ámbito. A lo largo del capítulo 3 se presentan los requisitos impuestos al sistema diseñado, así como las funcionalidades que se pueden extraer del mismo. El capítulo 4 recoge el diseño de la solución técnica, es decir, una descripción detallada del sistema así como detalles relevantes acerca de diferentes aspectos del mismo. Los apartados 5 y 6 ofrecen el plan de proyecto y las pruebas efectuadas (así como algunos resultados y consideraciones), respectivamente. Por último, los capítulos 7 y 8 exponen las conclusiones extraídas, las líneas futuras de trabajo y las posibles repercusiones que puede tener el sistema implementado en determinados entornos.

2. Estado del arte

A lo largo del presente capítulo se aporta una visión general del subtitulado. Abordando el tema desde diferentes puntos de vista se pretende que el lector sea capaz de entender de una forma general qué es el subtitulado, en qué consiste y la casuística asociada a su utilización, todo ello enmarcado dentro del ámbito de la accesibilidad a los medios audiovisuales.

Inicialmente se proporciona una somera descripción del subtitulado en general. Seguidamente, se profundiza en los tipos de subtitulado existentes enfatizando en el subtitulado en televisión, así como en las tecnologías de subtitulado que se utilizan actualmente y los problemas existentes. Tras aportar una descripción del marco regulatorio que se aplica en este ámbito, se explica de manera breve y concisa la tecnología del reconocimiento automático del habla, ya que es utilizada en numerosas ocasiones para subtitular y del mismo modo se hace uso de la misma en el presente proyecto. Por último, se comentan los trabajos relacionados desarrollados sobre esta disciplina, para tener constancia de los posibles avances actuales existentes.

2.1 Introducción

Definimos como técnicas de accesibilidad a aquel conjunto de prácticas que son utilizadas para posibilitar a las personas con una determinada discapacidad, generalmente visual o auditiva, pueda tener acceso a toda clase de contenido audiovisual.

Hay múltiples técnicas de accesibilidad, debido a la distinta naturaleza de cada una de las posibles discapacidades que pueden sufrir los seres humanos. Actualmente, las técnicas por excelencia son el subtitulado para sordos y la audiodescripción para invidentes. No obstante, existen otras bastante utilizadas, como por ejemplo la lengua de signos y la audionavegación. Algunas de las características de las dos primeras se comentan a continuación.

El subtitulado para sordos es algo análogo a la audiodescripción para los discapacitados auditivos. Es necesario diferenciar entre subtitulado simple, en el que únicamente se plasma en pantalla el texto que se dice, y el subtitulado para sordos. Éste se podría definir como una técnica de transcribir el audio a texto, y está pensada para personas que tengan algún tipo de discapacidad auditiva que les impida escuchar los sonidos que provienen de los productos audiovisuales. De esta manera, los subtítulos permiten al discapacitado, de alguna forma, “leer” todos los sonidos que suceden durante la acción. Por ello, los subtítulos consisten en una representación textual no exclusivamente de qué se dice, sino de cómo se dice (ya que el tono o si se enfatiza pueden ser variables relevantes en la acción que transcurre), así como de cualquier otro sonido que fuera necesario captar para permitir un perfecto entendimiento de la acción (como por ejemplo música, ruidos ambientales, etc.)

Por otro lado, la audiodescripción es un servicio destinado, principalmente, a personas invidentes o con alguna clase de discapacidad visual. Consiste en explicar de una forma breve, clara y concisa lo que sucede durante la acción en un producto audiovisual. Estos comentarios explicativos se introducen en las pausas del audio original del producto, con el fin de no entorpecer la comprensión del mismo. La idea es que en éstos no se explica exclusivamente qué sucede, sino cómo sucede, así como se aporta una descripción de cualquier elemento que pudiera resultar relevante: personajes, lugar de la escena, etc. Por consiguiente, la audiodescripción supone para las personas discapacitadas visuales un añadido que les posibilita el acceso a la televisión, el teatro, el cine o a cualquier otra arte visual de las que, de otro modo, no podrían disfrutar.

Como se puede comprobar, ambas técnicas suponen una traducción de la acción completa que se desarrolla en el producto audiovisual a audio y a texto, respectivamente. Con ellas se busca que todas las personas sean capaces de acceder a los contenidos audiovisuales a los que, sin ellas, a muchos les resultaría imposible. Por fortuna, en los últimos años ha crecido notablemente la concienciación social en este sentido. La sensibilización es un factor muy importante que permite obtener avances en estos ámbitos y esto, apoyado por una legislación gubernamental favorable, ha permitido que las técnicas de accesibilidad se encuentren en pleno apogeo dentro del ámbito de las sociedades más avanzadas.

2.2 Subtitulado en televisión

Los servicios de accesibilidad a la TDT, principalmente el subtitulado, la audiodescripción, la lengua de signos y la audionavegación, son elementos clave para el acceso a los contenidos audiovisuales que se emiten en la misma, que de otro modo no serían accesibles para más de un millón de personas en España debido a su discapacidad visual o auditiva. Adicionalmente, el servicio de subtitulado permite el acceso a la televisión a las personas con desconocimiento del idioma, así como a las personas que, por determinadas razones, se encuentran en circunstancias en las cuales les resulta imposible escuchar el audio de los programas de televisión.

El punto inicial de este apartado es la Ley 7/2010, de 31 de marzo, General de Comunicación Audiovisual (LGCA) [2], que entró en vigor el 1 de mayo de 2010 y que se explicará posteriormente a lo largo de este documento. Esta Ley marcó un punto de ruptura con el período anterior y supuso una nueva etapa en cuanto a las obligaciones de los operadores de televisión en materia de accesibilidad. No obstante, antes de entrar en vigor esta Ley, muchas cadenas de televisión se esforzaron en aplicar medidas que dotaran a sus emisiones de accesibilidad, pese a que no había ninguna clase de obligación al respecto. Tras la entrada en vigor de la LGCA la mayoría de las cadenas televisivas han ido aumentando sus porcentajes de subtitulado, puesto que deben llegar a la tasa mínima exigida por ley. En el resto de servicios de accesibilidad que se ofrecen, por contra, los datos no son tan favorables. En la siguiente tabla se plasman los porcentajes fijados que hay que alcanzar en los servicios de subtitulado,

audiodescripción y lengua de signos que los operadores de televisión (ya sean públicos o privados) deben cumplir de manera obligada.

Tabla 1: Obligaciones de accesibilidad en la LGCA

- La televisión privada:

	2010	*2011*	*2012*	*2013*
Subtitulado	25%	45%	65%	75%
Horas lengua signos	0,5	1	1,5	2
Horas audiodescripción	0,5	1	1,5	2

- La televisión pública:

	2010	*2011*	*2012*	*2013*
Subtitulado	25%	50%	70%	90%
Horas lengua signos	1	3	7	10
Horas audiodescripción	1	3	7	10

Para tener controlados los progresos que se consiguen al respecto, se está realizando una constante labor de monitorización y seguimiento de la accesibilidad en televisión. Esta labor de control de la accesibilidad adquirió mayor relevancia tras la aprobación de la LGCA, ya que se creó tras ello el Consejo Estatal de Medios Audiovisuales (CEMA), un organismo independiente encargado de controlar y verificar el cumplimiento de las directrices impuestas en dicha ley.

Queda claro pues que los índices de accesibilidad en los canales de televisión se encuentran en evaluación permanente. Entre muchas otras, una de las labores del CESyA es la de medir y verificar estos índices, de manera que gracias a las mediciones efectuadas en sus laboratorios, meticulosamente analizadas y contextualizadas, ha sido posible efectuar un seguimiento fiable de la evolución de las cadenas televisivas en materia de prestación de servicios de accesibilidad (principalmente subtitulado, audiodescripción y lengua de signos como se ha comentado con anterioridad) [3].

La siguiente figura muestra la evolución de los porcentajes de subtitulado en los últimos años en algunas de las principales cadenas de televisión en nuestro país.

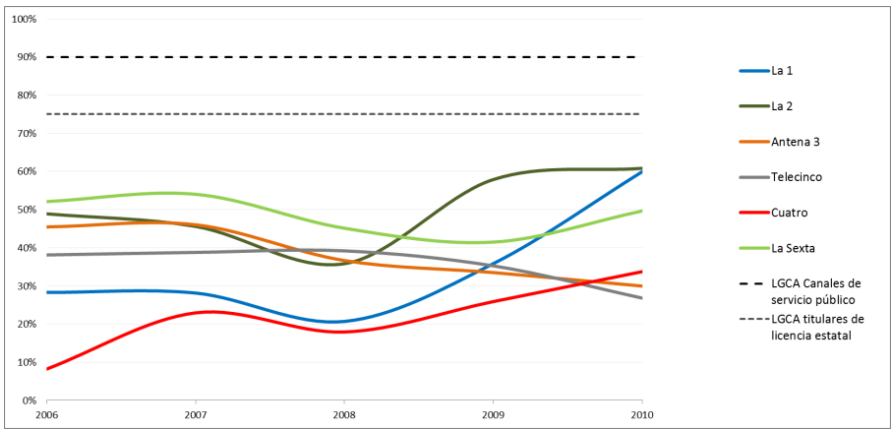


Figura 1: Evolución de los porcentajes de subtitulado 2006-2010(Fuente MITyC)

A la vista de los datos, se puede inferir fácilmente que los principales operadores de televisión se habían anticipado a la regulación, puesto que se extraen cifras de aproximadamente un 40% de subtitulado en algunos casos sin haber ningún tipo de legislación vigente en aquellos años que obligara a tales tasas. Sea como fuere, aún no se cumple con el porcentaje de subtitulado impuesto por la LGCA para el año 2013, por lo que actualmente las cadenas se encuentran trabajando activamente para cumplir con sus objetivos de accesibilidad.

El grado de madurez de la tecnología para emisión, producción y recepción de contenidos accesibles se considera satisfactorio para las personas con discapacidad auditiva. Se están desarrollando nuevas tecnologías de transmisión del subtitulado, que suponen una mejora en la presentación de los subtítulos. Los nuevos receptores de TDT deben cumplir las especificaciones técnicas pertinentes, propias de la recepción de la televisión digital, facilitando de esta forma el acceso al subtitulado de cada canal. Aunque los esfuerzos en este sentido son grandes, existen dos grandes dificultades desde un punto de vista tecnológico que suponen un grave problema para el servicio de subtitulado. Ambas están asociadas al servicio de subtitulado de los programas en directo, y es absolutamente necesario solventarlas.

- Por un lado, la calidad del subtitulado de los programas en directo no es óptima, debido a que hay una falta de sincronización bastante importante entre el audio y los subtítulos.
- Por otro, el coste que supone el subtitulado masivo de programas en directo.

Ya se han iniciado labores de investigación con este objetivo, pero se requiere mucho tiempo e inversión para alcanzar los niveles de coste y calidad impuestos en la LGCA. Concretamente este documento mostrará en capítulos posteriores cómo se trata de resolver esta situación, puesto que pretende reducir enormemente sendas dificultades de manera simultánea con la herramienta que se ha desarrollado para ello. El problema de la sincronización de subtítulos en directo afecta a todos, porque no existe un método hoy día que permita solucionarlo. Cadenas como la BBC o TVE emiten más de 100.000 y 40.000 horas de subtitulado respectivamente cada año, y los excesivos retardos son el

principal motivo de queja del colectivo de discapacitados auditivos, ya que impiden completamente seguir la acción de los programas emitidos.

En cuanto a la emisión de contenidos subtitulados en general, los porcentajes que satisfacen los canales más relevantes hasta ahora han cumplido con las tasas impuestas por la LGCA, pero dado el incremento de la proporción de emisiones en directo, es cada vez más complicado ir mejorando esas tasas. Esto es así porque la subtitulación de programas en directo es bastante más compleja dados los problemas expuestos, y además este servicio es uno de los más demandados por los usuarios. Por otro lado, el cumplimiento de la Ley en este sentido puede ser engañoso, debido a que muchas cadenas de televisión satisfacen los porcentajes mínimos impuestos debido a la alta proporción de emisiones en diferido. La siguiente tabla muestra las horas de directo y diferido que ofrecen muchas de las cadenas de televisión españolas existentes, así como la proporción de las mismas que está subtitulada [4].

Tabla 2: Horas subtituladas en directo y diferido, enero-febrero 2012 (datos CESyA)

Canal	Horas Directo	Horas Diferido	% Directo subt.	% Diferido subt.
24 Horas	74.44%	25.55%	61.94%	72.01%
Boing	X	100%	X	91.26%
Clan Tv	X	100%	X	99.4%
Cuatro	32.19%	67.80%	13.25%	71.68%
Disney Channel	X	100%	X	63.46%
Divinity	14.13%	85.86%	0%	71.24%
FDF	7.12%	92.87%	23.91%	77.38%
La 1	58.47%	40.13%	75.27%	82.17%
La 2	5.76%	94.23%	12.04%	76.19%
La Siete	19.25%	80.74%	0%	35.96%
La Sexta	37.90%	61.54%	34.90%	81.95%
La Sexta 3	16.56%	83.43%	0%	75.03%
Marca TV	44.85%	54.54%	59.58%	40%
MTV	X	100%	X	39.68%
Nitro	7.72%	91.50%	0%	51.78%
Telecinco	62.22%	37.77%	13.59%	59.42%
Teledeporte	29.82%	70.17%	100%	76.23%
Antena 3	33.38%	66.61%	0%	69.67%
Neox	12.5%	87.5%	0%	56.51%
Nova	0.59%	99.40%	0%	52.12%

Como se puede comprobar, las tasas de directo subtitulado son bastante inferiores a las de diferido. Solamente algunas, como La 1 o 24 horas muestran porcentajes altos de subtitulado, que se acercan bastante a los de diferido. No obstante, el resto de cadenas presentan, en general, muchas menos horas emitidas en directo por lo que subtitulando el diferido consiguen cumplir los objetivos impuestos en la LGCA en cuanto al subtitulado. Por tanto, aún queda un largo camino por recorrer, teniendo en cuenta que la tendencia actual es a aumentar el porcentaje de programación en directo.

2.3Tipos y tecnologías de subtitulado

En este epígrafe se realiza una distinción de los tipos de subtítulos existentes, así como de las tecnologías que se aplican en los diferentes escenarios de subtitulado.

2.3.1 Clasificación de los subtítulos

Los subtítulos se pueden clasificar de acuerdo a varios criterios, por lo que se han escogido varias clasificaciones y se exponen a continuación.

Uno de los parámetros que permiten efectuar una clasificación de los subtítulos es la audiencia que tendrá acceso a los mismos mientras éstos son mostrados:

- Se habla de **subtitulado cerrado** cuando los subtítulos son mostrados a una única persona, es decir, de forma que sólo pueden verse de manera individual.
- Por el contrario, cuando los subtítulos se muestran a varias personas de manera simultánea en una misma pantalla, se trata de **subtitulado abierto.**

Otra de las posibles clasificaciones es de acuerdo a la funcionalidad de los subtítulos, esto es, según el tipo de información que proporcionan al entorno audiovisual al que se corresponden:

- Los subtítulos son **narrativos** cuando corresponden a la transcripción del diálogo que está sucediendo durante la acción. Son, por tanto, los más habituales.
- Se trata de subtítulos **forzados** si aparecen para transcribir diálogos o elementos de la obra que se encuentran en idioma extranjero. Pueden aparecer traducidos o sin traducir en función del punto de vista que se pretenda transmitir en la obra al espectador.
- Si la información que muestran de la obra es completamente adicional, siendo por ejemplo algún apunte para introducir al contexto de la obra, se trata de subtítulos de **contenido** (por ejemplo los textos de inicio de la saga “La Guerra de las Galaxias”). A menudo incluso sustituyen a las imágenes.
- Son subtítulos **informativos** cuando añaden información adicional a la obra pero no forman parte de la misma (por ejemplo, comentarios del director).
- Los subtítulos se dice que son **cerrados** cuando muestran información contextual relevante, como puede ser algún ruido de fondo o algún ruido eventual. Muy utilizados en subtitulado para sordos.

Según el modo de distribución de los subtítulos junto con el medio audiovisual, se distinguen:

- Subtítulos **incrustados**. También llamados **permanentes**, se proporcionan insertados en la imagen a la que acompañan, por lo que no pueden ser separados de ella. Dado que son parte de la imagen misma no es necesario utilizar ningún elemento para su reproducción. A cambio, por el mismo motivo, no se pueden desactivar ni configurar (por ejemplo el tipo de fuente o el color). Así mismo,

impiden utilizar más de una fuente de subtitulado (por ejemplo para disponer de subtítulos en varios idiomas) y, al insertarlos en la imagen, podrían mermar la calidad de la misma.

- **Pre-renderizados** o **flotantes**. Se proporcionan en formato de imagen pero separados del stream de vídeo. De esta forma permiten tener diferentes fuentes de subtitulado para un vídeo, pero no modificar sus características de visualización. Además, el reproductor debe ser compatible con su formato y suelen ser codificados a baja calidad para reducir su tamaño en bytes.
- **Cerrados** o **aislados**. Se transmiten en un stream alternativo al del vídeo original y sin renderizar, esto es, como un conjunto de textos, marcas de tiempo y otros parámetros para que sean mostrados en el reproductor, que evidentemente debe ser compatible. Son ampliamente utilizados, debido a sus múltiples ventajas: tamaño en bytes reducido ya que se almacenan como texto y no como imágenes, sencilla creación y alta flexibilidad en cuanto a la visualización (incluso su formato de visualización se puede modificar durante la reproducción).

2.3.2 Métodos de subtitulado

Hay diferentes métodos de subtitulado en función de aquello que se desee subtitular. No obstante, la labor de subtitulado suele ser un proceso lento y laborioso, que requiere de unas herramientas específicas para poder ser llevado a cabo con garantías de éxito. Algunos de los métodos de subtitulado más utilizados son los siguientes:

- **Transcripción manual:** método clásico, en el que se transcribe manualmente a texto lo que se pretende plasmar en el subtítulo. Es un proceso sencillo pero muy trabajoso, con la ventaja de que los errores cometidos son mínimos. Es la técnica por excelencia para subtitular en diferido, ya que se dispone de tiempo suficiente para generar los subtítulos.
- **Estenotipia:** método en el que se utilizan teclados específicamente diseñados para lograr una escritura rápida, usando abreviaturas y otras técnicas. Suele ser utilizado en ocasiones para obtener las transcripciones en casos de tiempo real. A cambio de mayor velocidad y la baja probabilidad de error, es un proceso muy costoso. Hay muy pocos estenotipistas disponibles, por lo que es caro, y se suele utilizar en eventos importantes en directo, en los cuales se precisa de una calidad aceptable y el menor retardo posible.
- **Rehablado:** método orientado también a la generación de transcripciones en tiempo real. Se basa en personas que repiten la locución que se pretende transcribir (rehabladores) en un entorno controlado y exento de ruidos, provisto de los medios suficientes para entregar un sonido limpio a un sistema de reconocimiento automático de habla. Éste permite obtener de forma automática el texto dictado. Posteriormente, esos mismos rehabladores editan los subtítulos generados corrigiéndolos en caso de que sea necesario. Menos costoso que la estenotipia, también requiere de personas especializadas, y la tasa de obtención

de subtítulos es menos (es un proceso con más retardo). Se suele emplear para subtitular en directo en las situaciones que no cubre la estenotipia, dado que el coste de ésta es excesivo para satisfacer las leyes establecidas en cuanto a porcentajes de programas subtitulados.

2.3.3 Formatos de subtitulado

Existe una gran variedad de formatos para almacenar la información de subtitulado, adaptados a los existentes métodos de distribución y con numerosas opciones en función del sistema al que estén destinados. En cualquier caso, los subtítulos se pueden almacenar en archivos de texto plano o en forma de imágenes, como se ha comentado. En este segundo caso ocupan más, pero a cambio evitan al reproductor el renderizado del texto. Lo más importante de los subtítulos es el texto y las marcas de tiempo asociadas al mismo, y esos dos datos siempre se van a poder almacenar. Adicionalmente, dependiendo del formato, se puede introducir más información, como por ejemplo acerca del estilo (color, fuente, etc.). La elección del formato se efectuará en función de las necesidades que haya en cada situación concreta. Algunos de los formatos más habituales son:

- SubRip (.srt)
- MicroDVD (.sub)
- SubStation Alpha (.ssa)
- Universal Subtitle Format (.usf)
- Dvb-Sub

2.3.4 Escenarios de subtitulado

Una vez expuestos los aspectos técnicos generales más relevantes del subtitulado, se pueden describir someramente los entornos principales en los cuales se emplean todas las técnicas de subtitulado descritas. En función de la naturaleza temporal de la locución a transcribir se distinguen dos casos principalmente.

2.3.4.1 Diferido

El diferido es el caso en el que se pretende generar subtítulos a partir de un vídeo del que se dispone previamente. En otras palabras, se habla de diferido siempre que se vaya a subtitular algo pregrabado, que no sucede en ese mismo instante.

Este escenario es muy simple, porque no es necesario realizar ningún esfuerzo especial más allá del de transcribir a texto lo que se dice. La persona encargada de subtitular dispone del tiempo necesario para su trabajo y, en principio, sólo se tiene que preocupar de que los subtítulos estén dispuestos correctamente según los estándares establecidos. Para crear los subtítulos se utiliza la transcripción manual, no se requiere de ningún otro método más complejo o costoso. Como se ha comentado, subtitular de este modo es un trabajo sencillo pero algo lento y laborioso, por lo que el subtitulador generalmente utiliza herramientas que le facilitan el trabajo. Actualmente existen multitud de paquetes de software que abordan de diferentes modos la creación de subtítulos para material audiovisual. Hay por tanto una amplia oferta de aplicaciones tanto propietarias como

gratuitas, con características técnicas específicas en cada caso y que operan de manera diferente. Aún con esto, hay una serie de constantes que suelen cumplir debido a que son elementos de sentido común que en todos y cada uno de los casos facilitan y agilizan la labor de subtitulado. Algunas de ellas se enumeran a continuación:

- **Gestión de subtítulos:** Siempre suele haber en las herramientas diseñadas algún tipo de tabla que relacione los subtítulos con los tiempos de inicio y fin asociados, permitiendo así una visión global del conjunto de una forma extremadamente sencilla. Unido a poder cargar los subtítulos desde algún tipo de archivo compatible con el programa, se consigue que la edición sea muy sencilla y asequible.
- **Gestión de vídeo/audio:** Evidentemente cada subtítulo se relaciona con un fragmento de vídeo y audio que sucede entre el tiempo de inicio y el tiempo de fin expuestos en la tabla anterior. Por ello es muy útil disponer en la misma interface de usuario del vídeo, el audio o ambos para tener siempre presente qué subtítulo se quiere incluir en cada instante (no son elementos independientes en la mayoría de los casos). Así, en los programas se suele disponer de un visualizador del vídeo y de la forma de onda del audio, para poder cargar esta información. Con su ayuda, se puede discernir de una manera más exacta cuándo se dice lo que se quiere subtitular, y además es más sencillo e intuitivo el asignar tiempos de inicio y fin.
- **Gestión de tiempo:** Asociado al apartado anterior, la gestión de tiempo es esencial, dado que la sincronía entre el audio / vídeo y el subtítulo generado es necesario que sea lo más cuidada posible para asegurar la máxima comprensión por parte del usuario final. Generalmente se dispone de controles para delimitar los intervalos en los cuales se sitúa el texto del subtítulo de manera que la tarea sea lo más directa posible.
- **Estilo de subtítulos:** Es el aspecto visual que tendrán los mismos sobre la imagen. Las herramientas de subtitulado suelen ser bastante flexibles en este sentido, pudiéndose configurar y modificar parámetros de estilo como por ejemplo el color, la transparencia y la posición de la caja contenedora de los subtítulos o el color y la fuente de la letra empleada en ellos.
- **Guardado del subtítulo:** Todas las herramientas pueden grabar los subtítulos generados en alguno o varios de los formatos mencionados anteriormente. El tipo de formato escogido es importante según el caso concreto que se esté tratando, porque depende de diversos factores como por ejemplo el reproductor de destino (que puede ser compatible solo con algún tipo de formato determinado) o la tasa binaria de los canales de distribución (que si es baja se necesitarían archivos de subtítulos más ligeros, de menor tamaño).

Puesto que el diferido es un caso de baja complejidad, no es raro comprobar que las herramientas que se suelen utilizar son gratuitas y de unas características bastante más limitadas que las de sus contrapartidas de pago. En este escenario es obra del

subtitulador transcribir lo que se dice y lo que sucede, sincronizar con el vídeo que se está subtitulando mediante la asignación de tiempos manualmente y además asegurarse de que los subtítulos están correctamente particionados según la norma.

Una de las herramientas gratuitas más utilizadas en este sentido es Aegisub (Unix, Windows y Mac OS X), debido a su versatilidad en cuanto a sus capacidades para modificar el estilo de los subtítulos. No obstante hay muchas otras utilizadas, como Subtitle Workshop (Windows), Jubler (basado en Java, es mucho más sencillo, solo para pequeñas ediciones), Magpie (basado en Java también, sólo cubre necesidades básicas), Subtitle Editor (Linux) o Gaupol Subtitle Editor (Linux, con versión Windows, funcionalmente básico).

2.3.4.2 Directo

El directo es el escenario en el que es necesario generar los subtítulos a partir de una emisión que a priori se desconoce, por estar sucediendo en tiempo real. Este caso es bastante más complicado que el anterior, debido precisamente a que se va conociendo la información de forma paulatina, y es necesario transcribirla con la mayor celeridad posible. Esto provoca dos grandes problemas que son la base del presente texto:

- **Retardo:** El problema más grave existente hoy día en cuanto al subtitulado en directo, y el mayor motivo de queja por parte de los colectivos de discapacitados auditivos en nuestra sociedad. El tiempo que se tarda en generar los subtítulos en escenarios en directo es muy alto, excesivo. Como se ha mencionado, para subtitular en directo se aplican dos técnicas principalmente: estenotipia y rehablado. En el primer caso los retardos son menores (aunque siguen existiendo), pero el coste es excesivo[2]**.** En cambio en el segundo caso el coste es menor[3], a costa de más retardo incluso, debido a que todas las tareas – rehablado, edición, etc. – las realiza una única persona, sumado a los retardos inducidos por los ASR. Lo más utilizado es el rehablado, debido a que no es factible sostener el coste de la estenotipia, por lo que los subtítulos generados mantienen retrasos variables que pueden alcanzar más de 20 segundos. Esto obviamente es desastroso y no dificulta, sino que impide la comprensión de la emisión en muchísimas ocasiones.
- **Calidad de los subtítulos:** Asociado al problema anterior, dado que hay que generar subtítulos en el menor tiempo posible, la calidad de los mismos a menudo es baja, con alto porcentaje de fallos. Si bien este problema es menor (ya que errores puntuales en palabras no enturbian la comprensión global de los subtítulos), hay que tenerlo en cuenta también para tratar de solventarlo o atenuarlo en la medida de lo posible.

Por tanto, este es un escenario muy propenso a errores y problemas. Los encargados de generar los subtítulos utilizan a menudo herramientas que facilitan su labor, esta vez

[2] El coste de la estenotipia se encuentra en media entre 150 y 200€/hora.
[3] El coste del rehablado con edición incluida se encuentra entre los 8 y los 12€/minuto.

generalmente propietarias y por tanto de pago. Suelen encontrarse integradas en suites y sus precios no son bajos, pudiendo alcanzar hasta los 6000 € en algún caso. En cualquier caso estos conjuntos de programas son herramientas de edición profesional que permiten tratar con subtítulos tanto en diferido como en directo, de manera que resultan útiles en muchas ocasiones y son utilizadas por múltiples cadenas de televisión incluso a nivel mundial.

Algunas de las herramientas más empleadas son Swift (utilizada por la BBC, HBO o RAI, entre otras), FAB Subtitler (Antena 3, Canal +, Tele 5 o TVE), Caverna Tempo, SPOT Software o Sysmedia Wincaps Subtitling.

En [9] se hace un exhaustivo análisis del estado de la técnica y práctica del subtitulado de programas en directo mediante el uso de rehablado.

2.4 La problemática del subtitulado de programas en directo

Debido a que el trabajo que se ha desarrollado tiene como fin último conseguir un sistema que sea capaz de corregir los retardos existentes respecto al vídeo en escenarios en directo, es preciso profundizar más en la problemática existente en este ámbito.

Se ha explicado que los principales problemas existentes en relación a la generación de subtítulos en directo son los retardos y la tasa de errores. El más grave es el primero, puesto que es el que generalmente dificulta la comprensión de la acción, al presentarse al usuario la transcripción del discurso disociada de los elementos de comunicación no verbal presentes. Palabras erróneas aisladas, en general, no evitan que se entienda lo que está sucediendo. Estos retardos no son constantes, y se han efectuado estudios para constatarlo. Uno de ellos, efectuado por el CESyA [7], demuestra que dependiendo del tipo de programas de televisión que se consideren, estos retardos varían enormemente. Las siguientes figuras muestran los resultados de estas pruebas sobre dos tipos de entornos.

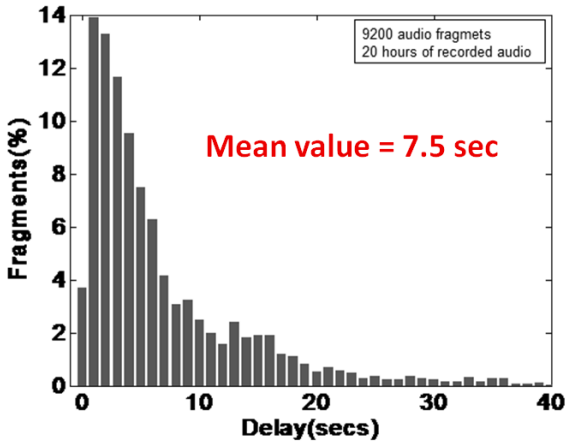


Figura 2: Retardos de los subtítulos en televisión

La figura 2 muestra los resultados obtenidos tras medir el retardo de los subtítulos obtenidos aplicando ASR directamente durante 20 horas de grabación de un canal de

televisión, sin hacer distinción alguna entre programas y con audio incontrolado, esto es, en el cual hay en ocasiones ruido o perturbaciones. Se puede apreciar que hay una gran variabilidad, superándose en muchas ocasiones los retardos de más de 20 segundos. La prueba inmediatamente siguiente es realizar el estudio de los retardos existentes en los subtítulos generados mediante rehablado, ya que se asemeja más a los entornos reales de televisión.

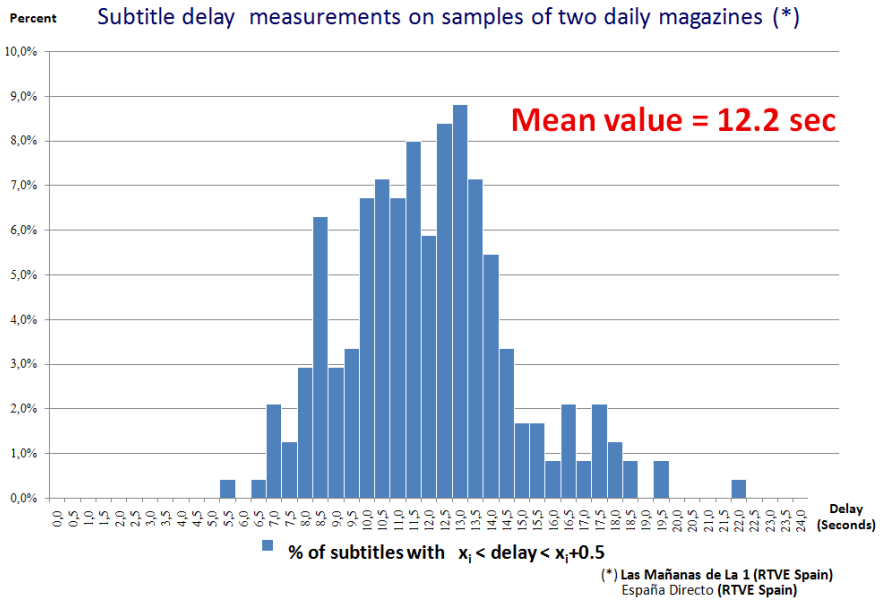


Figura 3: Retardos de los subtítulos de rehablado en televisión

En la figura anterior se plasman los retardos existentes entre el vídeo y los subtítulos generados mediante rehablado medidos en dos programas de RTVE. Del mismo modo, se comprueba que son muy variables, y encontrar una solución implica corregir individualmente todos y cada uno de los retardos, primero porque no existe un método que permita corregir los retardos globalmente, y segundo porque en tiempo real se van generando subtítulos cada pocos segundos, y hay que ir corrigiendo sus retardos según vayan siendo generados.

Por otro lado, de los trabajos relacionados (véase sección 2.7) se extrae también información interesante de cara al subtitulado en este tipo de entornos. La idea principal es que en estos escenarios, en los cuales un programa se digitaliza y se transmite al usuario, es posible compensar estos retardos introducidos, principalmente por la transcripción textual del audio. Además, debido a que la transmisión y la presentación de la información son dos pasos diferenciados, es posible conseguir una presentación sincronizada de los subtítulos (tras aplicar la corrección de los delays) con el vídeo y el audio retardándose estos últimos con respecto a los primeros en algún punto del proceso. Este proceso se encuentra descrito en la patente desarrollada por la UC3M, y reforzado por la posterior prueba de concepto creada, en la cual se consigue sincronizar la transcripción generada a partir del audio de un vídeo de prueba retrasando éste unos

segundos de manera que el tiempo que tarda el ASR en ofrecer el texto se ve compensado por el retardo introducido al vídeo manualmente.

2.5 Normativa

A continuación se procede a introducir el marco legislativo vigente sobre el que se sustenta todo lo explicado anteriormente.

2.5.1 Ley General de la Comunicación Audiovisual

La Ley 7/2010, de 31 de marzo, General de la Comunicación Audiovisual [2] supuso con su aparición un esfuerzo por parte del Estado de aunar y potenciar la normativa vigente válida entonces respecto a la industria audiovisual. En un mercado cambiante, en el cual hay una notable evolución de las tecnologías y de los modelos de negocio existentes, es necesario recoger una serie de normas que permitan regular a largo plazo y que protejan a los ciudadanos de posibles situaciones de dominancia de mercado. Esta Ley no sólo compendia la normativa anterior, sino que actualiza los aspectos que han sido modificados drásticamente y regula las nuevas situaciones que, hasta ahora, carecían de regulación.

La Ley General de la Comunicación Audiovisual se muestra como el principio básico a seguir tanto por el sector privado como por el público en cuanto a la prestación de servicios de televisión, radio y otros servicios interactivos. Por supuesto, la Ley procura mantener una situación de equilibrio entre las empresas pertenecientes al sector audiovisual, para garantizar la libre competencia siempre con el fin último de proteger al ciudadano y, además, promueve la igualdad de todas las personas que componen la sociedad, específicamente en lo referente a las discriminaciones de género y por motivo de las discapacidades.

En definitiva, la Ley General de la Comunicación Audiovisual conforma una normativa audiovisual moderna, coherente y liberalizadora, que a la vez que regula el sector refuerza los derechos de los ciudadanos y de la sociedad.

Esta Ley se estructura en seis Títulos, y se compone de 61 Artículos de diversa índole. Entre toda esa maraña de información legislativa, la más relevante desde el punto de vista del presente proyecto es la relacionada con la accesibilidad, esto es, los derechos de las personas con discapacidad. Esto es así porque es esta Ley la que obliga a los proveedores de servicio a dotar de accesibilidad a sus productos, en pos de permitir a los ciudadanos discapacitados acceder a los servicios audiovisuales de una manera completamente autónoma. Los porcentajes de subtitulado, audiodescripción y lengua de signos que han de ir alcanzándose en las cadenas de televisión a lo largo de los años venideros están recogidos en la LGCA, y teniendo en cuenta el notable incremento de las emisiones en directo respecto de las emisiones en diferido, es evidente que hay que encontrar una forma que permita ofrecer esas tasas sin que el coste sea excesivo ni la baja calidad del resultado final enturbie el esfuerzo empleado. A continuación se recogen los artículos y secciones más destacados de la Ley en materia de accesibilidad:

- Artículo 6: Derecho a una comunicación audiovisual transparente.
  Todos tienen derecho a conocer la identidad del prestador del servicio de comunicación audiovisual y la programación televisiva con antelación suficiente. Por ende, los apartados 3 y 6 de este artículo se recoge, respectivamente, que estas informaciones contenidas en páginas de internet, guías electrónicas de programas y otros medios de comunicación deben ser accesibles para las personas con discapacidad, así como los contenedores mismos de dicha información.
- Artículo 8: Los derechos de las personas con discapacidad.
  Las personas con discapacidad visual o auditiva tienen derecho a accesibilidad universal a la comunicación audiovisual, de acuerdo con las posibilidades tecnológicas. Así mismo, los discapacitados auditivos tienen derecho a un 75% de subtitulado en el servicio de televisión, al igual que las personas con discapacidad visual deben disponer de al menos dos horas semanales de televisión audiodescrita. Adicionalmente, tanto los poderes públicos como los prestadores de servicio fomentarán el disfrute de los medios audiovisuales para las personas con discapacidad, y siempre se procurará emitir desde un punto de vista respetuoso e inclusivo para las mismas.
- Artículo 18: Comunicaciones comerciales prohibidas en cualquiera de sus formas.
  Se prohíbe toda comunicación comercial que atente contra la dignidad humana o fomente la discriminación por motivos de sexo, raza, religión o creencia, discapacidad, edad u orientación sexual.
- Artículo 58: Infracciones graves.
  Entre las muchas infracciones graves posibles, en el epígrafe 4 se indica que es infracción grave incumplir los deberes de accesibilidad dispuestos en el artículo 8 (apartados 2 y 3) durante más de cinco días en un período de diez días consecutivos.
- Disposición transitoria quinta: Servicios de apoyo para las personas con discapacidad.
  Se indican las tasas de subtitulado, audiodescripción y lengua de signos que deben haberse alcanzado el 31 de diciembre de cada año, distinguiéndose entre el sector público y el privado. Se puede comprobar que para finales de 2013 el porcentaje de subtitulación requerido es del 90% y del 75% respectivamente.

En definitiva, este marco regulatorio que hay que cumplir supone a las cadenas de televisión en este caso un enorme esfuerzo, ya que las tasas de subtitulado que necesitan satisfacer son elevadas. Si además aumenta la proporción de directos en las emisiones, el esfuerzo y la inversión necesarios son aún mayores. No obstante, hay que cumplir estos requerimientos y, por consiguiente, investigar y tratar de generar herramientas para el gran reto que supone subtitular el directo con unas cotas de calidad relativamente aceptables.

2.5.2 Norma UNE de subtitulado

Una vez asimilado que es necesario moralmente, así como obligatorio por Ley subtitular en muchas ocasiones, es preciso seguir un estándar de subtitulado, que proporcione las directrices básicas de la generación de subtítulos. Para ello se dispone de la norma UNE 153010 de subtitulado [1], que recoge todos los aspectos referentes a la creación de subtítulos que es necesario cumplir.

Para lograr la igualdad de oportunidades y el completo acceso a la información defendido por la Ley General de Comunicación Audiovisual, hay que proporcionar un marco regulatorio que controle cada uno de los servicios de accesibilidad disponibles. En esta norma se trata el aspecto técnico únicamente del subtitulado en televisión, aunque muchos de los criterios expuestos son de sentido común y se aplican en muchos otros casos. La televisión supone una gran oportunidad para que las personas accedan al entretenimiento, la cultura y la información. Por ello, la manera más práctica de que las personas que mantienen una discapacidad auditiva tengan garantizado el acceso a dichos servicios es el subtitulado vía teletexto. Por otro lado, no sólo las personas sordas se benefician del subtitulado. Hay una serie de colectivos como pueden ser el público infantil o las personas con dificultades lectoras que se pueden beneficiar del subtitulado. Ahora bien, este servicio debe estar, de alguna manera, estandarizado, de modo que haya unos requisitos mínimos de calidad y homogeneidad establecidos en cuanto al subtitulado. La norma UNE tiene como premisas estas necesidades, y es fruto del consenso entre todas las personas que conforman el sector: administración, empresas, televisiones, profesionales del sector y usuarios, mostrándose especial atención a las opiniones y preferencias de las personas sordas que son, en último término, aquellas que más necesitan estas directrices. A continuación se recogen los aspectos más relevantes de la norma que se deben satisfacer, expuestos en el mismo orden que en ese documento. Para información más detallada o ejemplos explicativos, se adjunta la propia norma.

- **Aspectos generales:** Se dispone para generar subtítulos de un alfabeto compuesto por 128 caracteres, así como de un conjunto limitado de colores de letra y de fondo. Además, por línea se podrán tener como mucho hasta 35 – 37 caracteres.
- **Colores:** Hay combinaciones de colores que facilitan la legibilidad del subtítulo, como por ejemplo amarillo sobre negro, verde sobre negro o cian sobre negro. Cada personaje debe llevar asignado un color, que debe mantenerse siempre salgo que algo exija lo contrario. Lógicamente esta elección dependerá de la importancia del personaje. En directo, adicionalmente, hay que colocar antes el nombre del locutor si se usa una combinación de caracteres / fondo igual para todos, para poder distinguirlos.
- **Caracteres, líneas y ubicación:** El tamaño de los caracteres está normalizado a dos veces la altura de una línea de teletexto. El interlineado debe ser sencillo y como máximo debe haber dos líneas en pantalla generalmente, siempre en la parte inferior salvo cuando pudiera tapar información relevante. Así mismo, se

deben subtitular todos los sonidos posibles, en una línea y en la parte superior de la pantalla.

- **Paginación y división de los subtítulos:** El particionado de los subtítulos debe realizarse siguiendo una serie de criterios de comprensión y legibilidad. Por ejemplo, no se debe separar palabras en dos líneas. Hay que tratar de dividir la oración en líneas en función de los signos de puntuación (comas, puntos, etc.) o de las conjunciones. También se debe tener en cuenta que, a menudo, las pausas naturales del locutor marcan cuándo partir un subtítulo.
- **Tiempo de exposición de los subtítulos:** Para poder leer correctamente el texto en pantalla debe permanecer un tiempo dado, que se tratará de maximizar siempre que sea posible y el vídeo lo permita. Como mínimo, una línea corta debe permanecer expuesta 0.7 segundos.
- **Sincronismo de subtítulos:** Obviamente debe ser lo mejor posible entre la imagen y el texto aparecido en pantalla. Se debe procurar que aparezcan coincidiendo con el movimiento labial y la información sonora, para mejorar aún más la comprensión del conjunto y evitar confusiones. Aunque los personajes no aparezcan en pantalla también hay que subtitularlos.
- **Criterios ortográficos y gramaticales:** Es preciso que, como en cualquier texto, se cumplan las reglas ortográficas y gramaticales de la Real Academia Española. Si, por algún motivo, el locutor no cumple esas reglas no por un error puntual sino por ser un rasgo del mismo, no es necesario cumplirlas en el subtítulo, como transcripción misma de la locución. Debido a las limitaciones del alfabeto utilizado (128 caracteres), se deben escribir con letras las abreviaturas o caracteres que no estén en dicho alfabeto (por ejemplo el €). Además, los números decimales o mayores de 10 deben escribirse con cifras y no se pueden utilizar corchetes porque no se incluyen en el alfabeto. Se usarán paréntesis en su lugar.
- **Edición de subtítulos:** Los subtítulos deben ser literales en la medida de lo posible. De todos modos, cuando el orador locuta con una velocidad muy alta es necesario aplicar ciertas técnicas para no perder el sincronismo. En estos casos hay que utilizar siglas, evitar reiteraciones y utilizar formas cortas de personalidades u organismos conocidos. Siempre que sea posible se recomienda emitir por otra página de teletexto un subtitulado con palabras más coloquiales y asequibles, así como mayores tiempos de exposición, dedicado a las personas con posibles dificultades lecto-escritoras que pudieran considerar complicada la versión de subtítulos más literal.
- **Información contextual:** Se deben representar los efectos sonoros, en una línea y en la parte superior de la pantalla. Preferiblemente, hay que utilizar la descripción del sonido antes de la onomatopeya. Además, si el locutor se expresa con un tono determinado hay que representarlo también, en mayúsculas y entre paréntesis antes del texto que le corresponde para facilitar la comprensión del argumento.

- **Información que debe proporcionar el teletexto:** Se debe proporcionar información referente a los colores asignados a los personajes, así como un glosario de símbolos utilizados incluyendo su significado.

En el presente proyecto se ha tratado de seguir los criterios establecidos por esta norma en la medida de lo posible. Por un lado se ha cuidado de satisfacer el máximo de caracteres por línea y del número de líneas que hay que utilizar en los subtítulos. Por otro, el particionado tiene en cuenta los signos de puntuación, palabras como conjunciones y además trata de cumplir el tiempo mínimo de exposición en pantalla. Otras mejoras como los colores no se tienen en cuenta en la primera versión de la herramienta, cuyo objetivo es el de sincronizar con el audio de manera automática y no cuidar al máximo la estética de los subtítulos. No obstante hay toda una serie de mejoras admisibles que se comentarán en capítulos posteriores.

2.6 Reconocimiento automático del habla

Ya se ha comentado que una de las tecnologías aplicadas a la generación de subtítulos es la del reconocimiento automático del habla (*Authomatic Speech Recognition* - ASR). Estos sistemas logran obtener una transcripción textual a partir de la voz humana. Este proceso funciona del siguiente modo: a partir del audio de entrada, el cual proviene de una señal de voz de una persona dada, se extraen una serie de características de esa voz, los llamados *feature vectors* o vectores de características. En función de la tecnología concreta utilizada estas características pueden estar relacionadas con la potencia, la entropía, el dominio temporal, el dominio de las frecuencias o los coeficientes cepstrales de la señal original.

Por otro lado, en el ASR se dispone de un diccionario, modelos acústicos y modelos del lenguaje. Estos tres elementos son clave durante el proceso de reconocimiento. El modelo del lenguaje expone la estructura del lenguaje a nivel de palabra, es decir, qué palabra puede aparecer teniendo en cuenta las anteriores que han aparecido. El diccionario ofrece la pronunciación de las palabras del modelo del lenguaje. Esta pronunciación divide las palabras en secuencias de unidades, que son todas y cada una de ellas definidas en el modelo acústico. Éste último permite efectuar un mapeo entre una unidad acústica y un modelo oculto de Márkov (HMM) que pueda ser evaluado frente a un vector de características. Estos tres elementos trabajan en consonancia: cada palabra del diccionario se divide en unidades acústicas según el contexto, de las cuales se extrae su modelo de Márkov asociados y, con esta información y el modelo del lenguaje se genera un grafo de búsqueda que posibilitará ir procesando los vectores de características recibidos e ir determinando la secuencia de palabras transcrita. Evidentemente para ir obteniendo la secuencia más probable dentro de este grafo de búsqueda se aplica un algoritmo de búsqueda que no se encuentra restringido a ninguna implementación específica. De hecho, se pueden implementar algoritmos como el de Viterbi o el algoritmo de búsqueda A* para decidir (en general, algoritmos de búsqueda en grafos). Una vez se ha determinado la secuencia de palabras más probable, se devuelve como resultado.

Hay que distinguir entre el reconocimiento de habla orientado a comandos y el reconocimiento de habla continua. El primero es bastante más sencillo, puesto que hay, por así decirlo, un conjunto de palabras relativamente limitado que reconocer. Por ejemplo, para dictar números, hay diez posibles resultados. En este caso es más fácil que el reconocedor reconozca el comando que se dijo. El segundo es mucho más complejo, dada la enorme variabilidad de palabras. Además, debido al enorme número de combinaciones existentes, el proceso de reconocimiento es más lento y produce más errores.

La tecnología ASR es prometedora, teniendo en cuenta que permite reducir el coste del subtitulado ya que es capaz de transcribir de manera textual el audio. No obstante, el procesamiento de la señal es lento, debido al coste computacional asociado. Cuanto mayor sea la secuencia de unidades acústicas a reconocer, más grande es el grafo de búsqueda y más costoso es el proceso, en términos de tiempo y memoria de CPU. Estas unidades acústicas pueden ser palabras, fonemas o comúnmente trifonemas (tres fonemas consecutivos). Por otro lado el reconocimiento de habla no está exento de errores. Por un lado, las secuencias obtenidas como resultado son las más probables según el diccionario y los modelos del lenguaje de los que se dispone. Eso implica que no necesariamente tienen que ser correctas, hay cierta probabilidad de error. Adicionalmente, hay una dependencia del interlocutor a partir del cual se extraiga el audio. Los modelos del lenguaje son adaptativos y los sistemas ASR los van entrenando progresivamente para adaptarse a la voz que reciben. Como resultado, las tasas de error varían en función del usuario y del grado de entrenamiento del modelo. Por último, a menudo los interlocutores no hablan de una forma clara y precisa, sino que no pronuncian ciertos fonemas, se confunden o incluso generan ruidos como tos o la risa, que se cara a los ASR son elementos que distorsionan la señal y, por tanto, disminuyen la calidad del reconocimiento.

Debido a estos problemas, es necesario plantearse en qué escenarios es posible aplicar esta tecnología. Concretamente en entornos televisivos, lógicamente en directo porque el diferido se subtitula generalmente de manera manual, hay una serie de emisiones aptas para aplicar estas tecnologías. Entre ellas, destacan por ejemplo programas tales como los informativos, debates, programas de discusión o reportajes. Estas emisiones se suelen caracterizar por proporcionar un audio claro y una forma de comunicación pausada, con una dicción correcta y adecuada, sin opinar varias personas simultáneamente y con escaso ruido de fondo. De todos modos hay que tener muy en cuenta la calidad del audio, porque cualquier tipo de distorsión o ruido de fondo podría disminuir drásticamente la calidad del resultado. Se puede realizar una clasificación de los programas televisivos en función del ruido de fondo:

- **Programas con sonido claro** si no hay ruido de fondo.
- **Programas con sonido degradado** cuando hay perturbaciones en momentos puntuales. Por ejemplo una llamada telefónica en el programa.
- **Programas con música de fondo.**

- **Programas con ruido de fondo** si se puede oír ruido mecánico o de alguna otra clase (por ejemplo tráfico).
- **Programas con vocales de fondo**, cuando mientras habla el presentador hay otras voces escuchándose.

Según el tipo de programa, habrá que actuar de una forma u otra si se pretende utilizar la tecnología ASR con éxito.

Hay múltiples soluciones ya implementadas para el reconocimiento automático del habla, desde gratuitas hasta comerciales. Por un lado, el software gratuito más relevante es:

- **Xvoice:** Ofrece un reconocedor de habla para habla continua (dictado) y control por voz para muchas aplicaciones en Linux. Este sistema requiere el IBM ViaVoice Engine para funcionar, no siendo este último software libre y sin su licencia es imposible ejecutar Xvoice.
- **CMU Sphinx4:** Sphinx4 es actualmente un proyecto de código abierto. Incluye múltiples herramientas para los desarrolladores, como entrenadores, reconocedores, modelos acústicos, etc. Basado en Java, existen una librería (PocketSphinx) con las funciones básicas y un decodificador (Sphinx3) para reconocimiento de habla basados en C.
- **HTK3:** Kit de herramientas (Hidden Markov Model Toolkit) portable para construir y manipular modelos ocultos de Márkov, que son los utilizados durante el proceso de reconocimiento. Consiste en un conjunto de librerías y herramientas en C, que proveen de utilizades para el análisis del habla, el entrenamiento de los HMM y para la obtención de resultados. Pese a que es software libre, Microsoft retiene el copyright del código fuente del HTK original.

Por otro, destaca como herramienta de pago:

- **Nuance Dragon Naturally Speaking:** Dragon es un conjunto de herramientas orientadas al reconocimiento del habla desarrolladas y comercializadas por Nuance Communications, para Windows, soportando ambas versiones de 32 y 64 bits. Esta herramienta proporciona tres funcionalidades principalmente: dictado, síntesis de voz (TTS – text-to-spech) y entrada de comandos. Adicionalmente, aporta un Kit de desarrollo (un SDK) para utilizar en el ámbito del reconocimiento de habla basado en el motor de Dragon.

2.7 Trabajos relacionados

Por último, mencionar algunos trabajos desarrollados dentro del ámbito del subtitulado, así como de la sincronización de los subtítulos, que guardan relación con el presente proyecto. Gracias a dicho esfuerzo ha sido posible idear una solución válida a este problema, dado que ha permitido comprender más profundamente los complicados aspectos de esta problemática.

Entre todos los trabajos previos, destacan dos particularmente. Ambos están relacionados con la creación y sincronización de subtítulos en escenarios de televisión y tiempo real, por lo que cubren ampliamente el área en la que se inscribe el trabajo desarrollado. Sin estas bases, que se comentarán a continuación, habría sido imposible avanzar en una disciplina como esta.

El primero es una patente para la sincronización de subtítulos en directo [5]. Consiste en una gran cantidad de trabajo previo desarrollado por la Universidad Carlos III de Madrid, en el cual se define y se patenta el modelo asociado a todos los escenarios de subtítulos generados en tiempo real. Lógicamente este trabajo sienta las bases del proyecto actual, dado que se pretende conseguir un sistema cuyo fin último será ser utilizado en un escenario de televisión en directo para paliar los retardos asociados a la generación de subtítulos.

El segundo proyecto del cual bebe el presente trabajo es el "SYNCHRONIZED SUBTITLING IN LIVE TELEVISION. PROOF OF CONCEPT" [6]. Parcialmente financiado por France Telecom, se desarrolló entre 2010 y 2011. Este trabajo tuvo como objetivo principal crear una prueba de concepto que proporcionase subtitulado en tiempo real mediante técnicas de reconocimiento automático del habla[4]. El sistema que se implementó podía generar subtítulos a partir del audio de los canales de televisión, pudiendo reproducirlos de manera sincronizada con la emisión original (local o remotamente), así como emitirlos, aplicando un retardo respecto a la señal original en canales IPTV adicionales, seleccionables por los usuarios. En este proyecto se trabajaron dos temas principalmente. Por un lado, se investigó ampliamente en el campo de los reconocedores de habla (ASR) aplicados al subtitulado de programas de radiodifusión. Por otro, en la sincronización de subtítulos en determinados escenarios de televisión en directo. El sistema soportaba el subtitulado de audio en español.

Se puede afirmar que esta prueba de concepto es el antecesor del trabajo tratado en este documento. Esto es así porque en ella se transcribe el audio original directamente, tras lo cual se alinean los subtítulos generados a partir de esas transcripciones. El sistema constituye un primer paso en la demostración de la viabilidad de la sincronización individual de subtítulos, pero no es directamente aplicable a la mayoría de los programas de televisión en directo, pues la transcripción del audio original mediante ASR produce unas tasas de reconocimiento inaceptables para los usuarios de subtitulado. En el sistema implementado se realiza la operación de sincronización o alineamiento a partir de los subtítulos procedentes del rehablador y de la transcripción fidedigna, como se verá en capítulos posteriores. Hay otros trabajos que guardan relación con el tema que se trata, como por ejemplo un sistema que lanza los subtítulos de telediarios (extraídos de un guión) que se guía por un ASR aplicado al locutor. De esa manera se van lanzando los subtítulos de forma sincronizada [8].

[4] Para comprobar los resultados de la herramienta desarrollada, puede ver una demo dirigiéndose al siguiente enlace: http://www.youtube.com/watch?v=WzCENDbqf6I

3. Especificación de requisitos y funcionalidad

3.1 Introducción

A lo largo del presente capítulo se ofrece una descripción de los requisitos y funcionalidades del sistema de alineamiento y sincronización de subtítulos en escenarios de tiempo real con rehablado. Inicialmente se definen los requisitos que debe presentar el sistema, ya sean de usuario o requisitos técnicos más relacionados con las tecnologías y el contexto considerados. Después, se definen las funcionalidades de las que está dotado el sistema, esto es, los casos de uso para los cuales es válido el mismo.

3.2 Requisitos

Definimos requisito como la condición o capacidad que debe estar presente en un sistema o en algún componente del mismo para satisfacer un contrato, estándar, especificación u otro documento formal.

El sistema pretende solventar o, al menos, atenuar uno de los dos grandes problemas que existen a la hora de realizar subtitulado de emisiones en directo (el preocupante retardo que se produce). El otro problema, asociado al coste, no se trata directamente en el trabajo pero, como se verá en capítulos posteriores, es posible reducir el tiempo y el esfuerzo que se tarda en generar subtítulos en diferido. Debido a la naturaleza de estos problemas, no es trivial desarrollar una solución al efecto. No obstante, es algo perfectamente abordable pese a estar enormemente condicionado por el estado actual de la tecnología en cuanto al reconocimiento del habla (véase sección 2.6) y por el contexto que estamos considerando.

El punto de partida a la hora de determinar qué requisitos y funcionalidades ha de tener el sistema debe tener en cuenta dichos condicionantes. Por un lado, el escenario con el que se trata es el de rehablado, lo cual determina la forma en que debe funcionar la herramienta ya que debe estar adaptada a ese entorno. Por otro, hay que considerar las restricciones tecnológicas existentes, de las cuales se habla posteriormente (véase sección 4.5). Seguidamente, es necesario tener en cuenta los estándares y la normativa referente a la generación de subtítulos, debido a que la construcción de los mismos tiene que realizarse de acuerdo con dicha normativa. Por último, los usuarios. Es completamente crucial considerar qué buscan las personas discapacitadas auditivas, o cualquier otra persona que pueda utilizar este sistema (puesto que tiene otras funcionalidades), que son en última instancia los que se van a beneficiar de las utilidades de este sistema.

Tras analizar detenidamente estas variables, se han definido una serie de requisitos funcionales que el sistema debe satisfacer. Se exponen a continuación:

RU - 001	Nombre: Sistema de generación de subtítulos
Descripción: El sistema debe ser capaz de procesar un fragmento audiovisual y, a partir del mismo y utilizando técnicas de reconocimiento automático del habla y alineamiento de secuencias, obtener subtítulos coherentes temporalmente con éste.	

RU - 002	Nombre: Alineamiento de subtítulos
Descripción: El sistema debe ser capaz de alinear secuencias de palabras, esto es, dadas dos listas de palabras determinar cuáles faltan o son erróneas en cada una de ellas.	

RU - 003	Nombre: Sincronización de subtítulos
Descripción: El sistema debe ser capaz de, tras alinear, establecer la correspondencia entre los tiempos de referencia de las palabras alineadas, es decir, asignar los tiempos adecuados a las palabras que correspondan. Así mismo, el sistema ha de inferir de manera aproximada las marcas de tiempo no asignadas durante este proceso. En cualquier caso los tiempos deben referirse al reloj interno del vídeo.	

RU - 004	Nombre: Parser
Descripción: El sistema debe generar subtítulos automáticamente. Por ende, debe hacerlo de acuerdo al marco regulatorio establecido, al menos en cuanto al tamaño y particionado de los mismos. Estas características vienen descritas en la sección 2.4.2.	

RU - 005	Nombre: Formato de salida
Descripción: Los subtítulos generados por el sistema deben estar en formato .srt.	

RU – 006	Nombre: Modos de funcionamiento
Descripción: El sistema estará diseñado para soportar dos funcionalidades diferenciadas: diferido y directo.	

RU - 007	Nombre: Entradas
Descripción: El sistema deberá tener dos entradas de información: • La primera será común a ambas funcionalidades, y se tratará del audio original de lo que se debe subtitular. • La segunda podrá proceder de distintas fuentes en función del caso: rehablador, transcripción fidedigna del audio, transcripción degradada del mismo, etc.	

RU - 008	Nombre: Modo directo
Descripción: El sistema debe poder alinear secuencias y sincronizar en tiempo real estricto si opera en modo de funcionamiento “directo”.	

RU – 009	Nombre: Reconocimiento automático del habla
Descripción: El sistema debe utilizar un motor de reconocimiento, concretamente Dragon Naturally Speaking para transcribir el audio de entrada.	

RU – 010	Nombre: Audio
Descripción: El sistema debe funcionar en entornos con audio controlado, exentos de grandes perturbaciones o ruido de fondo.	

RU – 011	Nombre: Duración de los subtítulos
Descripción: La salida del sistema estará compuesta por un conjunto de subtítulos con tiempos de referencia sincronizados al vídeo y audio originales. Por ello, tras su generación debe comprobarse automáticamente que son legibles (duración adecuada), y modificar su duración en caso contrario.	

RU – 012	Nombre: Idioma
Descripción: El sistema debe funcionar en español tanto para el audio original como para las transcripciones alineadas.	

RU – 013	Nombre: Escalabilidad
Descripción: La herramienta debe estar diseñada para soportar en un futuro cambios, tales como diferentes idiomas, nuevas formas de generación de subtítulos o alineamiento si éstas fueran implementadas, de manera sencilla.	

3.3 Funcionalidades

A continuación se describen las funcionalidades principales del sistema, así como sus ventajas y entornos de operación más relevantes.

Como se ha explicado, este proyecto surge de la necesidad de solucionar, o bien disminuir notablemente al menos los graves problemas de retardo que sufre el subtitulado en directo. Si bien esto es cierto, tras realizar el análisis previo al diseño se pudo observar que si el diseño era el adecuado, se podría aumentar potencialmente la funcionalidad de la herramienta. De este modo sería posible ofrecer funcionalidades más allá de la que inicialmente se pretendía. Así podemos conseguir, por un lado, un sistema que puede resolver un grave problema que afecta a un sector importante de la sociedad y, por otro, una herramienta con una funcionalidad adicional pensada más para la persona que genera subtítulos que para la que los recibe. En definitiva, un sistema más completo y que es útil en ambos sentidos.

Por consiguiente, la herramienta proporciona dos funcionalidades claramente diferenciadas.

- Primero, una funcionalidad de subtitulado en directo (u online), en la cual la idea es reconocer el audio original de la emisión en directo mediante un ASR e irlo

comparando con la salida del ASR que utiliza el rehablador del programa. De esta forma, aplicando la técnica de *word spotting* (véase sección 4.2) se puede ir alineando y sincronizando subtítulos con el vídeo y audio originales con el fin de generar subtítulos con unos retardos bastante inferiores a los que se generan directamente a partir del rehablador.

- Segundo, una funcionalidad de subtitulado en diferido (u offline), la cual consiste en, a partir de un audio que proviene de un vídeo determinado, generar los subtítulos del mismo. Al igual que en el caso directo, el audio original se reconoce con un ASR, la diferencia es que en este caso la salida se compara con una transcripción en texto plano de lo que se dice en el vídeo (o bien se puede no aportar transcripción, aunque la calidad de los subtítulos es menor, lógicamente). Como resultado, se obtiene una versión de la transcripción alineada temporalmente con el audio original. El fin de esta funcionalidad es la de facilitar el proceso de subtitulado, una labor sencilla pero lenta y laboriosa.

Como se puede apreciar, ambas funcionalidades son en esencia la misma aplicada a dos escenarios diferentes. Por ello es una gran ventaja saber aplicar una misma técnica para afrontar dos problemas diferentes, ya que incrementa los casos de utilización.

A continuación se explican de una manera cualitativa las funcionalidades del sistema.

3.3.1 Funcionalidad directo

La funcionalidad de subtitulado directo (o de subtitulado en online) es la principal por la cual se definió el presente proyecto. Para comprender correctamente el funcionamiento de la misma es necesario explicar primero el contexto que estamos considerando:

Por tanto nos encontramos en un entorno en el cual la información de que se dispone generalmente es la siguiente:

- Audio y vídeo originales, con los cuales hay que sincronizar los subtítulos creados mediante técnicas de rehablado.
- Subtítulos con retardo variable[5], pero bien formados ya que el rehablador se ocupa no sólo de generarlos sino de editarlos. Nótese que estos subtítulos no coinciden necesariamente con lo que se dice en el vídeo, ya que el rehablador también interpreta y abrevia lo que se dice (lo cual supone mayor complicación, como se explica posteriormente).

La funcionalidad directo de la herramienta pretende conseguir, a partir de esta información, una versión de los subtítulos con un retardo nulo o muy inferior al existente de forma inicial, a costa de retrasar ligeramente el audio/vídeo del programa de televisión un poco, el tiempo suficiente para realizar la sincronización de los subtítulos originados en tiempo real. El funcionamiento es sencillo conceptualmente:

[5] El valor medio en programas como “Las mañanas de la 1” o “España Directo” (RTVE) es de unos 12.2 segundos, pero en muchas ocasiones los retardos superan los 20 segundos. Para más información, véase capítulo 2.4.

Por un lado, el audio original (que es lo que proporciona los tiempos de referencia con los cuales sincronizar el texto) es procesado por el ASR, de modo que la herramienta va obteniendo y bufferizando las transcripciones resultantes. Estas transcripciones se generan como bloques de palabras con tiempos de referencia veraces, ya que provienen del audio original. Es necesario destacar que muchas palabras son erróneamente reconocidas por el ASR y muchas oraciones incongruentes, debido a la calidad de las transcripciones generadas tras el reconocimiento. Esta cantidad de errores se produce porque los perfiles de voz que se utilizan son genéricos, no entrenados (ya que se desconoce en un principio la naturaleza del audio –si los perfiles fueran entrenados el comportamiento sería peor– y además otro problema clave es el nivel de ruido en el audio de los programas –cuanto mayor grado de solapamiento de voces o perturbaciones, peores prestaciones del ASR–). Por otro, van llegando a la herramienta los subtítulos generados por el rehablador, correctamente formados y editados pero con tiempos de referencia incorrectos, con retardo significativo y variable respecto al audio original. Una vez llegan, se van comparando por bloques con las transcripciones previamente almacenadas. El motor de alineamiento se encarga de aplicar los algoritmos de alineamiento basados en la mencionada técnica de *word spotting*, de modo que se establecen correspondencias entre palabras generadas por el rehablador y palabras provenientes de la transcripción obtenida mediante el ASR. Posteriormente, aplicando la algoritmia diseñada en este proyecto, se asignan los tiempos de las citadas palabras y se infieren los tiempos de referencia de las palabras de las que no se ha encontrado correspondencia en el alineamiento. Por último, una vez se dispone de las palabras con tiempos de referencia adecuados, el motor de generación de subtítulos (el Parser) se encarga de particionar los subtítulos de acuerdo con la norma UNE 153010 descrita en la sección 2.4.2, de una forma estética y coherente.

3.3.2 Funcionalidad diferido

La funcionalidad de subtitulado diferido (o subtitulado offline) es una funcionalidad derivada de la anterior, en cuanto que no era un objetivo a conseguir del presente proyecto pero, tras analizar el problema y diseñar la solución, se consideró que suponía un añadido atractivo y sencillo de implementar, ya que la arquitectura es similar al poder definirse un subsistema común a ambas.

Esta funcionalidad está orientada a facilitar el proceso de subtitulado. Actualmente es un proceso lento y laborioso, que se realiza manualmente. De media, es necesario invertir de 8 a 10 minutos para subtitular cada minuto de vídeo. Este proceso conlleva, primero, que la persona encargada transcriba lo que se dice en el vídeo a subtitular. Segundo, que esa persona vaya escuchando el audio del vídeo y sincronizando la transcripción. Tercero, generar los subtítulos de acuerdo a la norma y de una forma legible y estética. Si se automatiza alguno de estos procesos el encargado de subtitular gana mucho tiempo, lo cual se traduce en una disminución directa de los costes.

En este caso los datos de partida de los que se dispone son los siguientes:

- Audio y vídeo originales, con los cuales hay que generar los subtítulos sincronizados.
- Transcripción de lo que se dice en el vídeo. También es posible funcionar sin ésta aunque, evidentemente, la calidad final de los subtítulos será peor.

La funcionalidad diferido de la herramienta pretende construir, a partir de la transcripción, unos subtítulos sincronizados con el vídeo. El funcionamiento es el siguiente:

De forma completamente análoga al caso directo, el audio original del vídeo se procesa mediante un ASR, obteniéndose transcripciones del mismo en forma de bloques de palabras (con errores) cuyos tiempos son veraces. En este caso, en función del vídeo se pueden utilizar perfiles entrenados o no, dependiendo de cada escenario concreto. Por ejemplo si se trata de una charla de una persona de la cual se conoce el modelo acústico, es mejor utilizarlo. En caso de intervenciones de múltiples individuos es mejor utilizar un perfil neutro. Dicho esto, si no se ha aportado ninguna transcripción no hay nada con lo que alinear estas transcripciones, con lo cual únicamente el motor de generación de subtítulos construye los subtítulos con las mismas. Lógicamente, en este caso los subtítulos contendrán errores léxicos, gramaticales y/o de coherencia, debido a la probabilidad de error del ASR al procesar habla continua. En cambio, si se ha aportado una transcripción literal completa de lo que se dice durante el vídeo, el motor de alineamiento funcionará de forma similar al caso directo, aplicando el *word spotting* entre las transcripciones del audio obtenidas mediante ASR y el texto literal, para posteriormente asignar e inferir tiempos del mismo modo, esto es, se asignando los tiempos a las palabras que fueron alineadas e infiriendo las marcas de tiempo de aquellas que no lo fueron. Por último, se generará un fichero de subtítulos en formato .srt para poder realizar una reproducción sincronizada del programa subtitulado.

4. Diseño de la solución técnica

4.1 Introducción

Durante esta sección del documento se presenta el diseño completo de la solución planteada para satisfacer los requisitos y ajustarse a las funcionalidades descritas a lo largo del apartado anterior. Primero se trata en profundidad el alineamiento de secuencias, ya que la solución escogida está basada en la técnica de *word spotting* y, por ende, se necesitan estos algoritmos. Para continuar se indican las herramientas que se utilizarán para desarrollar el sistema así como las que se utilizan en el mismo. Seguidamente se ofrece una visión de las posibles restricciones que pueda tener la herramienta y, por último, se describe detalladamente el sistema: la arquitectura, los elementos e interfaces que la componen y el funcionamiento.

4.2 Planteamiento de la solución

Antes de abordar el diseño de la aplicación es necesario determinar la forma en que se va a resolver el problema general. Analizando la naturaleza del mismo se deduce que una de las posibles soluciones más realizables es la de aplicar *word spotting* y después recalcular individualmente los tiempos de referencia originales de las palabras que sean necesarias.

La idea es la siguiente: lo que se pide es generar una versión sincronizada de los subtítulos, de acuerdo a un audio determinado. Para ello, teniendo en cuenta el contexto de operación descrito en la sección anterior, se tienen subtítulos generados por un rehablador (recuérdese, bien formados en general pero con tiempos de referencia erróneos debido al retardo producido en la creación de las transcripciones del audio en directo) y, paralelamente, las transcripciones del audio obtenidas a partir de un ASR, con errores pero con tiempos de referencia respecto al video correctos. Por lo tanto es una buena solución tratar de establecer la correspondencia entre ambos conjuntos de palabras. De este modo, si se consiguen determinar ciertas palabras como correspondientes en ambos, como se sabe que los tiempos de referencia de las transcritas son correctos, es posible asignarlos a las que provienen del rehablado, para corregir posteriormente su retardo. Evidentemente todas las palabras generadas por el rehablador para las que no ha sido posible corregir su tiempo de referencia siguen manteniendo el retardo, por lo que es necesario un mecanismo adicional que permita recalcular sus tiempos de presentación.

El *word spotting* es una técnica que se utiliza para buscar, dentro de un flujo de audio, determinadas palabras o textos. Se pretende aplicar en este proyecto porque lo que se intenta para corregir los retardos es buscar palabras clave dentro del audio original que permitan conocer qué tiempos de referencia son fiables o no. Una vez esto es conocido, el resto de tiempos se infiere a partir de los primeros.

Por consiguiente el proceso se realiza en dos fases: una de alineamiento, donde se aplica el mencionado *word spotting* y se determinan las palabras coincidentes; otra de asignación individual de tiempos, en la cual se copian los tiempos de referencia de las palabras con correspondencia y a partir de los mismos se calculan los tiempos de palabra de las que no tienen. Posteriormente se generan los subtítulos de acuerdo a las restricciones establecidas, tales como la normativa vigente.

Una vez realizado este proceso, se puede efectuar una reproducción sincronizada del vídeo con los subtítulos generados, tras retrasar éste globalmente unos segundos, como se comentó en la sección 3.3.1.

Recuérdese además que se plantean dos modos de funcionamiento para el sistema. Como se puede observar, tanto el funcionamiento como la arquitectura del sistema para aplicar ambas funcionalidades son idénticos, por lo que es interesante prestar ambos servicios de una manera unificada. Adicionalmente, puesto que abordar directamente la funcionalidad directo es complicado, se plantea primero implementar la funcionalidad diferido y luego extenderla. Abordar el problema secuencialmente supone una ventaja a la hora de diseñar y planificar.

4.3 Alineamiento de secuencias

Una vez decidido que se van a buscar palabras clave en el audio original que coincidan con las rehabladas, el siguiente paso es determinar de qué forma se va a efectuar esa correspondencia. En el presente proyecto se utiliza un algoritmo que permite alinear secuencias y resolver ese problema con una notable tasa de acierto.

En informática, la programación dinámica se define como una técnica que permite, discretizando y secuenciando, optimizar problemas complejos. Es un método utilizado para reducir la complejidad y el tiempo de procesado de un algoritmo basándose en la premisa de que éste se puede subdividir en subproblemas superpuestos más sencillos. En otras palabras, si se divide un problema grande en subproblemas que se resuelven de forma óptima, en conjunto la solución es óptima. Los subproblemas se dividen a su vez del mismo modo, hasta llegar a problemas triviales. Esta técnica fue inventada por Richard Bellman en 1953.

Los algoritmos de alineamiento de secuencias son algoritmos que, utilizando técnicas de programación dinámica, se aplican en el ámbito de la bioinformática para alinear secuencias proteicas o de nucleótidos. El alineamiento de secuencias es una forma de comparar dos o más cadenas de ADN, ARN o proteínas y resaltar las zonas en las que son similares. Estas cadenas se representan con letras y, tras alinearlas, se determinan las similitudes y las diferencias. Por ejemplo:

Supongamos que disponemos de dos secuencias de ADN, que son *GTCCATG* y *ACTCTG*. Se necesita conocer si hay nucleótidos que difieren o faltan en alguna de ellas, con lo cual se alinean, resultando lo siguiente

_GTCCA**TG**
AC**TC__TG**

Resaltados se encuentran los nucleótidos coincidentes. El resto están, o bien cambiados o ausentes en alguna de las cadenas. Como se puede apreciar el alineamiento permite saber exactamente en qué puntos las cadenas coinciden. La aplicación de esto al presente proyecto es evidente: en el caso que se trata las secuencias son de palabras y no de caracteres, pero adaptando algún algoritmo de este tipo se puede ejecutar el *word spotting* que se pretende.

Hay diferentes tipos de alineamiento y por tanto de algoritmos para alinear. La principal distinción es entre alineamiento local y global de secuencias:

- Los alineamientos locales son útiles cuando las secuencias son diferentes pero con regiones de parecido alto entre ellas. Los algoritmos de alineamiento local ofrecen un resultado que es óptimo en este caso, pero no son capaces de alinear secuencias completas.
- Los alineamientos globales tratan de alinear las secuencias al completo, es decir, cada elemento de cada secuencia. Son útiles en casos de secuencias similares en longitud pero de las que no se estima en principio que hay regiones parecidas entre ellas. Los algoritmos de alineamiento global aportan una solución más general, en la cual se da el alineamiento más probable en toda la secuencia (y no a fragmentos como en el caso de alineamiento local). Son algo más lentos computacionalmente pero el resultado en términos generales es más estable, simplemente porque alinean las secuencias totalmente.
- Existen métodos híbridos que combinan ambas técnicas.

Por otro lado el alineamiento no tiene por qué aplicarse exclusivamente a pares de secuencias. Es perfectamente posible alinear conjuntos de n secuencias, aunque la complejidad de los algoritmos aumenta enormemente y, en la práctica, no se suelen alinear más de 3 ó 4 secuencias a la vez. En el caso que nos ocupa el alineamiento es de exclusivamente un par.

Hay múltiples ejemplos representativos de estos tipos de alineamiento. Se han estudiado los más relevantes para poder distinguir el más adecuado, y se detallan a continuación.

4.3.1 Algoritmo de Smith-Waterman

El algoritmo de Smith-Waterman[6] (conocido así por sus autores, pues fue propuesto por Temple Smith y Michael Waterman en 1981 [11]) es un conocido ejemplo de algoritmo utilizado para alineamiento local de secuencias. Debido a que está basado en

[6] Nota: Si el lector busca una explicación del funcionamiento del algoritmo más intuitiva, puede acceder a la siguiente dirección: http://docencia.ac.upc.edu/master/AMPP/slides/ampp_sw_presentation.pdf

programación dinámica, garantiza encontrar el alineamiento local óptimo de acuerdo con un sistema de puntuación determinado.

El funcionamiento del algoritmo es conforme a dos pasos:

- Generación de la matriz de puntuaciones de forma recursiva.
- Recorrido inverso de la matriz de acuerdo a ciertos criterios para ir encontrando la secuencia local alineada óptima.

A continuación se describe brevemente el proceso, para que se pueda comprender perfectamente el funcionamiento del algoritmo, así como para que sea posible compararlo con el resto de una forma concisa y precisa.

Sean s_i, t_j los elementos iésimo y jotaésimo de las secuencias s y t (s secuencia vertical, t secuencia horizontal)

Sea M la matriz de puntuación

Sea g la puntuación de alinear s_i con hueco o un hueco con t_j (g es negativo)

Sea $S(s_i, t_j)$ la puntuación de alinear el elemento s_i con el elemento t_j

Figura 4: Nomenclatura matemática de los algoritmos

Primero hay que colocar las secuencias (de longitudes m y n) y generar la matriz de puntuación (que será de orden n x m). A esta matriz se le aplica una condición inicial y después se va rellenando de forma recursiva:

-Paso 1: Se aplica la condición inicial, poniéndose a cero las primera fila y la primera columna.

$$M[i,0] = M[0,j] = 0$$

-Paso 2: Se rellena la matriz recursivamente. La celda $M[i,j]$ se rellena en función de lo que valieran las celdas adyacentes izquierda, superior y diagonal superior izquierda, para ir anotando en la matriz la puntuación almacenada de todos los posibles alineamientos entre las secuencias.

$$M[i,j] = max \begin{Bmatrix} 0 \\ M[i-1,j-1] + S(s_i, t_j) \\ M[i-1,j] + g \\ M[i,j-1] + g \end{Bmatrix}$$

-Paso 3: Se localiza la celda de la matriz con valor más alto. Dado que si se acierta se suma cierta puntuación y si se falla se resta, es evidente que el camino de la matriz con mayor puntuación es el de la secuencia alineada más probable.

-Paso 4: Como hay que recorrer la matriz en orden inverso, se recorrerá desde esa celda hasta que lleguemos a un 0 o al inicio de la matriz, lo cual implica que el alineamiento local ha concluido. La elección del recorrido por la matriz determina, en cada iteración, lo que hay en las secuencias alineadas (ya sea hueco en alguna de ellas, acierto o sustitución, se determina lo más probable). Moverse en diagonal implica que se asignan como alineados s_i y t_j, mientras que si el camino es hacia arriba o hacia la izquierda por la matriz es que hay un hueco en la secuencia t o s respectivamente.

La complejidad de este algoritmo es del orden de $O(mn)$, con lo cual si las secuencias son muy largas el espacio necesario y el coste computacional pueden volverse bastante elevados.

4.3.2 Algoritmo de Needleman-Wunsch

El algoritmo de Needleman-Wunsch [7] [10] es un ejemplo representativo del alineamiento global de secuencias. Utiliza también programación dinámica y fue ideado por Saul Needleman y Christian Wunsch, en 1970. El funcionamiento es muy similar al caso anterior, y garantiza obtener el alineamiento óptimo entre dos secuencias.

Funciona del mismo modo que el algoritmo anterior, primero rellenando una matriz de forma recursiva para, posteriormente, recorrerla en sentido inverso y obtener el alineamiento más probable. Dadas las similitudes, se utilizará la misma nomenclatura que para el caso de Smith-Waterman. A continuación se explica el funcionamiento y las diferencias con respecto a dicho algoritmo:

-Paso 1: Se aplica la condición inicial, poniendo la primera fila y la primera columna a un valor determinado, en función de la penalización por hueco. A diferencia del caso anterior, ahora sí se tienen valores negativos de la matriz, y esto servirá para alinear completamente.

$$M[i,0] = i \times g;\ M[0,j] = j \times g$$

-Paso 2: Se rellena la matriz recursivamente, de forma completamente análoga al caso anterior. El criterio que se aplica es el siguiente:

$$M[i,j] = max\begin{Bmatrix} M[i-1,j-1] + S(s_i,t_j) \\ M[i-1,j] + g \\ M[i,j-1] + g \end{Bmatrix}$$

-Paso 3: Equivalente al paso 4 anterior. Se comienza sobre la última celda de la matriz (la de abajo a la derecha) y se recorre la matriz completamente, hasta la celda superior izquierda. Al igual que antes, hay tres posibles movimientos: diagonal, izquierda y

[7] Nota: Para una explicación más visual del algoritmo, el lector puede remitirse a la siguiente dirección: http://www.ludwig.edu.au/course/lectures2005/Likic.pdf

arriba e implican alineamiento (correcto o erróneo), hueco en la secuencia t o en la secuencia s.

La diferencia fundamental es que mientras que en el caso anterior únicamente se recorría un fragmento de la matriz para alinear un fragmento de las secuencias, en este se recorre la matriz entera y se determina un alineamiento completo. Esto es tremendamente útil para aplicarlo a la solución planteada en este proyecto, porque no serviría conocer un alineamiento parcial de las secuencias de palabras que se obtengan. Se necesita una información más completa de qué palabras faltan y dónde faltan, y cuales se han alineado. Adicionalmente, la complejidad de este algoritmo es la misma que la del anterior, $O(mn)$, pero la solución se ha diseñado para que las secuencias de palabras que se vayan alineando sean de tamaño controlado, de forma que el coste computacional de las operaciones no sea excesivo y, por tanto, los retardos de procesamiento sean despreciables.

Se ha tomado, por tanto, este algoritmo como referencia para implementar la solución. No obstante, el diseño prevé la inclusión de nuevas formas de alinear y se ha construido el sistema de manera que se pueda configurar a gusto del usuario el algoritmo de alineamiento. Por esto, sería suficiente con implementar otro algoritmo cualquiera y se podría utilizar sin ningún problema en la herramienta.

Como se ha explicado, estos algoritmos se utilizan para alinear secuencias que, desde un punto de vista informático, se pueden considerar caracteres. Una complicación adicional del presente sistema es que requiere alinear palabras, esto es, conjuntos de caracteres, con lo cual es mucho más complejo definir las puntuaciones por acierto (ya que hay distintos grados de acierto: que las palabras coincidan completamente o que coincida solamente la raíz, por ejemplo) y las penalizaciones por hueco. Tras realizar diversas pruebas se han definido ciertas distancias (distancia entre palabras, entre secuencias de palabras, etc) y establecido ciertos umbrales de decisión (a partir de qué distancia de palabras se consideran alineadas, por ejemplo) que funcionan aceptablemente bien, pero ajustar tal cantidad de parámetros requiere de una enorme cantidad de pruebas, para poder generalizar razonablemente. Se ha diseñado un algoritmo que permite obtener el grado de parecido a nivel de palabra y a nivel de secuencia para poder realizar estos ajustes. A nivel de palabra es más simple porque se compara carácter a carácter pero a nivel de secuencia es más complicado, porque el algoritmo debe tener en cuenta que al menos una de las secuencias es potencialmente imprecisa y con errores (debido a que está obtenida de la transcripción del ASR).

4.4 Herramientas utilizadas

A continuación se presentan las tecnologías escogidas para implementar el sistema. Así mismo, se exponen las decisiones que han llevado a la elección de dichas tecnologías.

Microsoft .NET Framework

La plataforma .NET[8] de Microsoft es un componente software que se puede añadir al sistema operativo Windows. Provee un extenso conjunto de soluciones predefinidas para cubrir las necesidades generales de la programación de aplicaciones haciendo énfasis en la transparencia de redes y la independencia de la plataforma hardware.

.NET se ha tomado como plataforma de desarrollo y como lenguaje de programación se ha elegido C#. Su elección se fundamenta en sus amplias funcionalidades, el número de bibliotecas ya implementadas que ofrece y la capacidad de generar código de manera rápida; .NET provee lo que se conoce como código administrado, esto es, un entorno que aporta servicios automáticos al código que se ejecuta. Los principales servicios que ofrece son un cargador de clases para localizar clases en tiempo de ejecución, un recolector de basura, un motor para gestionar la seguridad del código, etc. Soporta más de 20 lenguajes de programación y es posible desarrollar cualquier tipo de aplicación en la plataforma con cualquiera de ellos. Algunos de los lenguajes desarrollados son C#, Visual Basic, C++, J#, Phyton, Fortran, Cobol, PowerBuilder…

Dragon Naturally Speaking

Como se ha visto en la sección 2.5, Dragon Naturally Speaking es una suite de herramientas desarrolladas por Nuance, de pago y para Windows, que implementan mecanismos para el reconocimiento automático de habla, entre otras funcionalidades. Una de las bases fundamentales del proyecto es la de que, como en el entorno televisivo se utilizan reconocedores automáticos de habla, utilizar como base un ASR para después poder extrapolar la solución a un entorno real.

Entre las múltiples opciones existentes se ha elegido DNS por tres motivos fundamentales. Primero, porque ya dispone de opciones para el reconocimiento de habla continua, algo básico y fundamental para este proyecto. Por otro lado, porque en el grupo de trabajo donde se ha desempeñado el trabajo (el CESyA) se dispone de licencia para poder utilizar este programa. En caso contrario, perfectamente se podría haber utilizado cualquiera de las utilidades de software libre existentes, pero ello implicaría una mayor cantidad de trabajo en materia de reconocimiento: generar los modelos, entrenarlos, etc. Con Dragon este proceso es muy rápido y sencillo, y en un breve período de tiempo (unos 10 minutos) se puede disponer de un modelo entrenado que esté listo para el dictado. Por último, porque ofrece una API que permite al desarrollador que lo utilice obtener información detallada de las transcripciones obtenidas, como el tiempo del audio correspondiente (algo de vital importancia en el

[8] http://www.microsoft.com/NET/

presente proyecto), o la probabilidad de que una palabra sea cierta (una información que se podría utilizar en posteriores revisiones de los algoritmos si fuese necesario).

4.5 Consideraciones tecnológicas

A continuación se muestran las posibles restricciones desde un punto de vista principalmente tecnológico que es necesario considerar para entender por completo el funcionamiento de la herramienta.

Pese a que esta solución es útil y puede mejorar notablemente los problemas expuestos en capítulos anteriores, hay que tener en cuenta que es también imperfecta. Imperfecta porque la tecnología actual no permite mejores prestaciones en el ámbito del reconocimiento de voz. Imperfecta porque hay que determinar de forma automática qué palabras están alineadas o no, lo cual implica determinar umbrales de decisión al efecto con la consecuente probabilidad de error derivada de ello. Imperfecta porque la inferencia de tiempos se realiza en algunos casos sin ningún otro tipo de información más que los tiempos de referencia de palabras sueltas. En definitiva, hay elementos mejorables con el tiempo si se realizan pruebas y revisiones y se optimizan los algoritmos de alineamiento y asignación de tiempos, y hay otros para los que es necesario esperar a que la tecnología avance si se pretende obtener una mejora en la calidad.

Hay, por tanto dos grandes restricciones principalmente, que se describen a continuación:

- El problema de los ASR. La calidad de los reconocedores automáticos de habla actualmente no es óptima debido a las condiciones ambientales. En otras palabras, para garantizar tasas de acierto razonables requieren de condiciones adecuadas de calidad de audio, esto es, entornos libres de ruido o con las mínimas perturbaciones posibles. Además se precisa de entrenamiento del sistema para maximizar la calidad del reconocimiento. Es cierto que el reconocimiento de comandos está a la orden del día y que incluso los teléfonos móviles funcionan bien en este sentido. Pero reconocer un conjunto discreto de comandos dista mucho de reconocer habla continua. Las posibilidades aumentan enormemente en este último caso, así como la complejidad y dificultad del reconocimiento. Lo que se necesita es que el reconocimiento de habla continua cumpla unos requisitos mínimos de precisión, pero la tasa de aciertos de un ASR utilizando un perfil inadecuado puede llegar a ser ínfima, incluso de menos del 15%. Trabajar con un elemento de tan poca fiabilidad implica que haya una determinada probabilidad de error en el sistema desarrollado, que no se puede evitar. No obstante, en el algoritmo central del sistema, tampoco es necesario que el ASR aplicado al audio original del programa de televisión acierte por completo, pues los algoritmos de inferencia de tiempos generados corrigen en cierta medida los posibles errores en la asignación de tiempos. De esta forma se ha comprobado que con tasas de acierto no muy elevadas del ASR (incluso del

30-40%) la calidad del resultado es más que aceptable. Es decir que si se mejora el postprocesado que se realiza tras el cálculo de tiempos (véase sección 4.6.2.1.3) es posible paliar a grandes rasgos estas imperfecciones, aunque es imposible que el resultado esté perfectamente sincronizado si la calidad de los ASR no mejora, por el simple motivo de que los tiempos se infieren automáticamente según unas directrices pero las personas no están programadas y, por ejemplo, el ritmo de palabras que pronuncian no es constante.

- Los algoritmos de alineamiento. Los algoritmos de alineamiento que se pueden aplicar (descritos algunos anteriormente, véase sección 4.3) funcionan de manera óptima en el campo de la bioinformática, porque se ejecutan sobre secuencias discretas. Por ejemplo si se pretende obtener el alineamiento de dos cadenas de ADN, desde el punto de vista de algoritmia se están alineando dos cadenas de caracteres cuyo alfabeto está compuesto por cuatro elementos distintos (ya que el ADN está formado por secuencias de cuatro nucleótidos diferentes). Al ejecutarse el algoritmo se trata de determinar si el elemento a alinear es o no es el mismo en ambas secuencias, y no hay otra opción posible. Es decir, al alinear *AG**A*** y *TAG**G***, la letra sombreada de la primera secuencia no es la sombreada de la secuencia dos (o bien es y se alinea ese elemento o bien no lo es y habrá hueco en alguna secuencia o sustitución).

  Es mucho más simple que cuando se trabaja con palabras: en este caso ni hay un alfabeto discreto (la variabilidad de palabras es enorme) ni el criterio es tan simple como si un elemento es o no el mismo en ambas secuencias. Veamos un ejemplo: al alinear *hola me **llamo*** y *hola me **llama*** está claro que ambas secuencias de palabras deben estar alineadas, pero las palabras sombreadas no son idénticas. Este ejemplo tan sencillo sirve para ilustrar la idea de que haya umbrales de decisión. Cuando las palabras a alinear son palabras con la misma raíz y diferente morfema, o bien si se trata de diferentes formas verbales, es bastante probable que se trate de algún error del ASR y que haya que considerarlas alineadas, pero las palabras difieren. En el presente sistema se determina de una forma completamente automática si las palabras deben ir alineadas o no, en función del porcentaje de parecido entre ellas. Por esto, aparecen errores inherentes a la automatización de las decisiones, que únicamente se puede esperar minimizar tras realizar numerosos estudios y pruebas y ajustar los umbrales de decisión de una forma óptima.

4.6 Arquitectura y funcionamiento del sistema

Como se ha planteado anteriormente, el sistema se ha diseñado para poder ser utilizado en múltiples modos de funcionamiento. Debido a que ambas funcionalidades poseen muchos elementos comunes, es más coherente plantear el sistema de una forma conjunta, es decir, determinar en qué aspectos ambas funcionalidades coinciden y en cuáles difieren. De ese modo se evita realizar dos implementaciones paralelas, y se reutilizan múltiples subsistemas que son similares para ambas partes.

Por ello en esta sección se plantea, en primer lugar, el modelado que se aplica a los datos para entender la arquitectura. En segunda instancia, la arquitectura común del sistema, explicándose los componentes de los que consta, las interfaces entre dichos componentes y el funcionamiento de los mismos. Posteriormente se repasan los elementos propios de cada arquitectura concreta, para obtener una visión completa de cada una de ellas y notar las diferencias entre las mismas.

4.6.1 Modelado de la información

Para poder desarrollar el sistema es necesario plantear una base, esto es, definir de qué manera se modela la información de que se dispone para poder trabajar sobre ella. No se pretende detallar en profundidad el código implementado, sino explicar los elementos que constituyen la base de las operaciones que se realizan durante el proceso de alineamiento para poder comprender el proceso completo.

Se definen cuatro elementos principales que transcurren por el sistema a modo de flujo de información, que son los siguientes:

- **Word:** Los objetos Word son las palabras propiamente dichas, con sus tiempos de inicio y fin de palabra asociados. A partir de este punto, hablar de Word implica hablar de la unidad básica de información que transcurre por el sistema, consistente en una cadena de caracteres (la palabra en sí) y el tiempo de inicio y el tiempo de fin asociados a dicha cadena de caracteres. Se ha determinado que los signos de puntuación pertenezcan al campo de texto de este tipo de objetos, y a su vez se ha tenido en cuenta para que la existencia de estos caracteres no alfabéticos no supusiese un problema al tratar de alinear.
- **Block:** Agrupaciones de un número dado de Words. Un Block no es más que una lista de cualquier número de objetos de tipo Word. Esta agrupación se utiliza y tiene sentido en determinadas partes del sistema por simplicidad respecto a los elementos que se explican a continuación.
- **Line:** Un objeto Line es una agrupación de objetos Word que pertenecen a la misma línea de subtítulo. Dicho de otro modo, una línea está constituida por varias palabras, pero no un número arbitrario sino un número determinado, que depende del tamaño de las propias palabras y de la posición de las mismas (debido a la norma de subtitulado). El objeto Line sería una particularización del objeto Block, donde la agrupación no es de un número arbitrario de Words.
- **Caption:** Un objeto Caption es una agrupación de objetos Line. Es el subtítulo propiamente dicho que se muestra en pantalla sincronizado con la imagen. Caption es a Line lo que Line a Word (al igual que una línea es un conjunto determinado de palabras, un Caption es un conjunto determinado de líneas). Ese número es configurable, no obstante se utilizan Captions de 2 líneas en este caso porque la norma de subtitulado así lo contempla.

Así, según la parte del sistema que se contemple tendremos unos elementos u otros, en función del nivel de abstracción necesario en cada caso. Por ejemplo, el alineamiento se

realiza a nivel de Word mientras que más adelante se procesan Captions en bloque para realizar corrección de tiempos de presentación de subtítulos.

Por otro lado, se observa que la estructura Word-Line-Caption es jerárquica. Esto también se ha diseñado de esta forma a propósito, porque simplifica en gran medida la implementación de determinadas partes del sistema (por ejemplo la generación y escritura de los archivos de subtítulos a partir de los Caption obtenidos). Además, aporta coherencia ya que realmente es el orden natural del subtitulado. Cualquier subtítulo (Caption) que se compruebe presenta un conjunto de bloques de palabras con tiempos de inicio y fin, cada uno de ellos constituidos por varias líneas (Line), las cuales están formadas, obviamente, por palabras (Word).

4.6.2 Arquitectura común

La mayor parte de elementos del sistema son comunes y su funcionamiento es idéntico en ambas arquitecturas. Así mismo, la información que transcurre por dichos elementos también es de características similares. Esto se ha forzado de esta manera por el motivo anteriormente mencionado: plantear la solución de una manera general, que permita abordar ambos casos de forma análoga.

En la siguiente figura se muestra la arquitectura general del sistema:

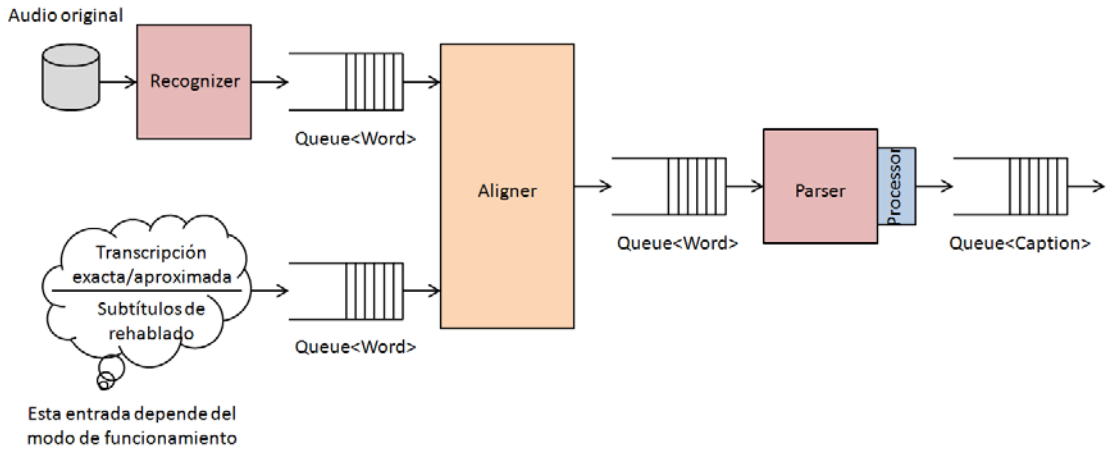


Figura 5: Arquitectura principal del sistema

A la vista de la figura anterior se comprueba que el sistema está subdividido en varias entidades funcionales, que engloban los algoritmos y las propiedades necesarias que se van a utilizar durante el transcurso de la información desde la entrada al sistema hasta la generación de subtítulos sincronizados. Hay una serie de partes principales desde un punto de vista funcional, que realizan labores bien diferenciadas. Todas ellas trabajan en consonancia en diferentes hilos de ejecución, de modo que el acceso a todas y cada una de las colas que actúan a modo de interfaces entre los elementos funcionales está controlado y se efectúa de una manera ordenada y concurrente. Además, todas estas operaciones se realizan en un hilo de ejecución diferente al principal precisamente para

que se tenga en todo momento control sobre el sistema completo por si se precisa realizar cualquier otra acción relevante, como es previsible. Se enumeran a continuación tanto los elementos más importantes de la arquitectura común como las interfaces de comunicaciones definidas entre los mismos:

4.6.2.1 Elementos funcionales

4.6.2.1.1 Reconocedor de voz

El elemento **Recognizer** constituye el reconocedor de voz que se utiliza durante la ejecución del sistema. Hay multitud de situaciones diferentes y en la elección del reconocedor de voz y del modelo acústico a emplear reside una parte importante del éxito durante el alineamiento. El reconocedor de voz toma el flujo de audio original que proviene del programa de televisión o del vídeo que se pretenda subtitular y efectúa el proceso de reconocimiento del mismo para transformarlo en texto.

De cualquier manera, el reconocedor de voz es un elemento con alta tasa de errores y más en esta parte del sistema, donde el ruido asociado al audio reduce las prestaciones del mismo. Así mismo, se utilizarán generalmente perfiles neutrales para el reconocimiento, puesto que se puede comprobar que a mayor grado de entrenamiento de un ASR por parte de un individuo, mejores prestaciones al utilizarlo éste y peores al utilizarlo los demás. En otras palabras, un perfil genérico puede funcionar peor que un perfil entrenado pero generaliza mejor y, en principio, en entornos de televisión no se dispone de perfiles acústicos de las personas que hablan ni se conoce a priori el orden en que lo hacen. En caso de subtitular vídeos en diferido tampoco es factible utilizar un perfil entrenado si en el vídeo intervienen personas diferentes, como es lógico.

No obstante, tampoco es necesario que la transcripción generada sea ideal, sino que es admisible cierta tasa de errores. Este elemento generará un flujo de objetos Word cuyo texto asociado sea, en multitud de ocasiones, erróneo (debido a errores en el reconocimiento) pero cuyos tiempos de inicio y fin de palabra asociados sean correctos y precisos, ya que el ASR es capaz de proporcionar dichos tiempos referidos al audio reconocido, que es el audio que proviene del propio directo y, por tanto, es a tiempo real.

4.6.2.1.2 Motor de alineamiento

De entre todos los elementos que constituyen el sistema, el más importante desde un punto de vista funcional es el **Aligner**. Todos los demás son absolutamente necesarios, por supuesto, pero este es el elemento central que determina y utiliza los algoritmos que definen el presente proyecto. Como se observa en la figura 5, este elemento recibe dos flujos de objetos Word y genera uno solo. El primer flujo de entrada es el que proviene del elemento funcional anterior, el reconocedor de voz y, por consiguiente, corresponde a la transcripción aproximada del audio original. El segundo flujo proviene de, o bien un fichero (de texto o subtítulos) en el caso de diferido o bien de los subtítulos generados mediante rehablado en el caso de directo. Los detalles se concretan en secciones posteriores para cada uno de los casos.

El **Aligner** realiza dos labores principales y de vital importancia. Por un lado, realiza el alineamiento de los flujos de entrada, asignando tiempos como se explica a continuación. Por otro lado, efectúa la inferencia de los tiempos no asignados durante el proceso anterior, de modo que a la salida se obtenga un flujo de palabras con tiempos correctamente asignados. Estos tiempos son los que se utilizan más tarde para reproducir el vídeo y el audio junto con los subtítulos de forma síncrona.

Como se ha visto anteriormente, cuanto mayor sea el número de elementos para alinear mayor es la probabilidad de encontrar el alineamiento óptimo. Por ello, se bufferizan las palabras generadas por el **Recognizer** hasta que hay un número admisible (este parámetro es configurable). Una vez se dispone de un número aceptable de objetos Word con tiempos correctos en la primera de las entradas, se extrae de la segunda un número similar de elementos para alinear. El segundo flujo de entrada está compuesto por palabras con texto correcto y tiempos de referencia erróneos o directamente inexistentes (en diferido se parte de una transcripción que si está en texto plano por ejemplo no contiene ninguna marca de tiempo). Para mantener la coherencia se ha procurado que el funcionamiento sea similar en el caso directo y el diferido: como en el primero el rehablador genera ráfagas (bloques) de palabras, en diferido se extraen bloques de palabras cada cierto tiempo, también a ráfagas, y se va alineando de esta forma por bloques. Como en el modo de funcionamiento de diferido se dispone del archivo completo de transcripción se podrían cargar inicialmente todas las palabras de dicho archivo, pero emulando el funcionamiento del directo todo es más consistente y el proceso es análogo en ambos casos. Esa es por tanto la opción elegida para los escenarios de alineamiento en diferido.

Una vez se dispone de dos bloques de palabras, uno con tiempos adecuados y texto erróneo y otro con tiempos retardados (o sin tiempos) y texto correcto, se procede a alinear e inferir. Ambos procesos se realizan de forma casi simultánea, esto es, se alinea una ráfaga, se asignan los tiempos de las palabras alineadas, se infieren los tiempos de las palabras no alineadas en esa ráfaga y se inserta la misma en el flujo de salida. Esos pasos se aplican a cada par de ráfagas que entran al motor de alineamiento, con la única excepción de las palabras finales de un vídeo en diferido, donde es probable (se ha comprobado experimentalmente) que finalice el reconocimiento de voz y algunas palabras del final se hayan quedado sin alinear. En este caso se infieren sus tiempos de una forma ligeramente diferente.

- Alineamiento y asignación de tiempos: En el presente proyecto se utiliza para el alineamiento el algoritmo de Needleman-Wunsch adaptado (véase sección 4.3.2), pero es posible escoger el algoritmo para alinear, en caso de tenerlo implementado. El alineamiento de dos ráfagas de n palabras se realiza a dos niveles, y en ambos casos se utiliza este algoritmo. Por un lado, a nivel de Block, esto es, el alineamiento de las ráfagas en sí. A este nivel, se trata de determinar qué palabras deben estar alineadas y cuáles no. Para ello se han definido umbrales de decisión, para determinar de manera automática si dos

palabras deben considerarse alineadas o no. Estos umbrales son configurables y se han establecido en torno al 45 – 50% de parecido de palabras, porcentajes que en las pruebas efectuadas dan unos resultados aceptables. Este grado de parecido entre palabras se obtiene aplicando el mismo algoritmo a nivel de Word, comparando esta vez secuencias de letras y obteniendo la distancia entre las mismas, entendiéndose por distancia al número de inserciones, sustituciones y borrados que hay que aplicar a la primera secuencia para transformarla en la segunda. A este nivel el algoritmo no falla debido a los motivos expuestos en la sección citada, ya que el alfabeto es un conjunto discreto de caracteres.

Una vez se ha determinado qué palabras están alineadas y cuáles no[9], se obtiene una pre-secuencia de salida a partir de la secuencia obtenida del segundo flujo de entrada (texto correcto) sobre la cual se copian los tiempos correctos de las palabras del primer flujo que se han alineado.

Los umbrales de decisión más importantes a determinar para el correcto funcionamiento del sistema son la máxima distancia admisible entre palabras y el mínimo alineamiento admisible entre secuencias. El primero indica cuál es la máxima distancia normalizada que pueden haber entre dos palabras w1 y w2 para considerarlas alineadas, y esta distancia se obtiene de la forma:

$$D(w1, w2) = \frac{(i+s+b)}{max\{long\,(w1), long\,(w2)\}}\ ,\ 0 \leq D \leq 1$$

El segundo indica cuál es el porcentaje mínimo de alineamiento admisible entre dos secuencias de palabras para considerar que el algoritmo no pierde la convergencia. Se obtiene aplicando la siguiente fórmula:

$$D_s(s1|s2) = 1 - \frac{número\ de\ alineamientos\ de\ s1\ sobre\ s2}{long(s1)}\ , 0 \leq D_s \leq 1$$

Por ejemplo, s1 es “Hola me” y s2 “Hola me llamo”, s1 se alinea por completo (la distancia es cero). En cambio si la distancia es alta, s1 se ha alineado mal sobre s2.

- Inferencia de tiempos: La secuencia anterior está compuesta por texto correcto y tiempos adecuados en las palabras que fueron alineadas en el paso anterior. A partir de esos tiempos de referencia fiables se obtienen los tiempos restantes, aplicando un algoritmo que consiste en tomar los tiempos de inicio de una palabra alineada y de fin de la siguiente palabra alineada y repartir esa diferencia entre todas las palabras intermedias. Veamos un ejemplo:

  Hola *me llamo* ***Alejandro***

  Supóngase que *Hola* comienza en el instante t_0 y que *Alejandro* finaliza en $t_0 + \Delta$, y que ambas palabras fueron alineadas. Dado que ambas palabras fueron alineadas, es evidente que el conjunto de las cuatro palabras comience en t_0 y finalice en $t_0 + \Delta$. Por tanto, es una aproximación razonable repartir la

[9] Para ver algún ejemplo en este sentido, véase sección 6.2.1 de este documento.

diferencia de tiempos (Δ) entre las cuatro palabras. De esa forma se consigue una secuencia de palabras con texto correcto y tiempos adecuados, en mayor o menor medida.

El algoritmo de alineamiento, debido a causas como el ruido, la baja calidad del primer flujo de entrada o a otras causas indeterminadas puede perder la convergencia. A veces sucede que, dados los umbrales establecidos, al tratar de alinear dos secuencias el resultado no sea válido. Es decir que no haya alineamiento para ese par de secuencias. En esas ocasiones las palabras que las componen se bufferizan y se vuelve a aplicar el algoritmo de alineamiento a la siguiente pareja de secuencias de entrada. Si esta imposibilidad de alinear se prolonga en el tiempo, se dice que se ha perdido la convergencia. Esta situación se prevé en el sistema, pudiéndose recuperar éste generalmente tras unos segundos, aunque los resultados generados referentes a ese momento de la ejecución tendrán más errores, como es previsible. Como se ve, este escenario es muy cambiante y pueden aparecer problemas imprevistos, pero eso se contempla en este sistema. Las palabras a partir de las cuales se infieren los tiempos de las no alineadas se mantienen siempre como referencia en el motor de alineamiento. De esta forma, si de una ráfaga no se puede alinear absolutamente nada, aún es posible inferir sus tiempos aunque, evidentemente, el error es mucho mayor. Un problema que aparece cuando se pierde la convergencia son los errores de sincronismo. Al inferir tiempos de un número muy alto de palabras sin haberlas alineado, es muy probable que no se repartan de forma adecuada debido a que la persona que habla en el audio original y a partir de la cual se transcribe no siempre lo hace con la misma velocidad, dicción o tono. Esta variabilidad provoca que, si inferimos tiempos cuando se va alineando (ejemplo expuesto anteriormente), la aproximación planteada es razonable. En cambio, si lo hacemos cuando se ha perdido la convergencia, hay que repartir un tiempo Δ entre un número muy elevado de palabras, produciéndose algunos subtítulos después muy extendidos temporalmente y otros demasiado cortos. No obstante, esto también se trata de corregir posteriormente.

Por último, la inferencia de tiempos es ligeramente similar en las palabras del final en diferido, dado que es posible que se acabe el primer flujo (el del **Recognizer**) y del segundo queden algunas. En este caso se calculan los tiempos de éstas a partir de la última palabra de referencia del motor de alineamiento, estimando la duración de cada palabra hacia delante.

Una vez realizados ambos procesos para cada par de ráfagas de entrada, se inserta la correspondiente secuencia de palabras resultante en la cola de salida, generándose de este modo un flujo de palabras con texto y tiempos de referencia adecuados.

4.6.2.1.3 Generador de subtítulos

El **Parser** es otro elemento crucial dentro del núcleo del sistema completo. Es el elemento encargado de transformar un flujo de palabras en subtítulos propiamente dichos.

El generador de subtítulos toma el flujo de salida de la etapa anterior y genera los subtítulos de acuerdo con la norma AENOR de subtitulado. Esto implica distribuir las palabras en líneas y las líneas en subtítulos. Este paso puede parecer simple pero no es en absoluto trivial. Se podrían generar subtítulos de una forma arbitraria, teniendo en cuenta simplemente que el máximo de caracteres por línea es de 37 según dice la norma, pero esos subtítulos no estarían bien formados, y uno de los objetivos del presente proyecto es lograr una generación de subtítulos automática que esté, en la medida de lo posible, de acuerdo a la norma y los criterios de calidad y coherencia estipulados mediante los estándares de subtitulado. Hay una serie de criterios de procesamiento del lenguaje que se trata de cumplir en la medida de lo posible, como por ejemplo que los signos de puntuación (tales como la coma, el punto y coma, el punto, dos puntos, etc.) deben ir si es posible y coherente al final de las líneas, o que las palabras como preposiciones o conjunciones deben estar situadas al principio de las mismas, pero las líneas deben tener además un tamaño adecuado sin exceder el máximo permitido, entre otras consideraciones. Teniendo en cuenta esas restricciones, se ha desarrollado un subsistema capaz de, dado un flujo de palabras, ir asignándolo en subtítulos siguiendo en la mayoría de los casos estas premisas.

De este modo, se van tomando palabras del flujo de entrada y se van estructurando en posibles líneas, según los criterios de procesamiento del lenguaje natural; por ejemplo empezar las líneas de los subtítulos por palabras como conjunciones y finalizar en signos de puntuación, que además en ningún caso pueden superar el tamaño máximo permitido en caracteres. Una vez realizado esto, se tiene un conjunto de líneas de tamaños completamente variables. Mediante un segundo paso se compactan dichas líneas si es posible (esto es, se juntan varias líneas si se puede), de forma que se genera un conjunto de líneas en general de mayor tamaño, de modo que se sigue cumpliendo todos los criterios: tamaño menor o igual que 37 caracteres y comienzo y fin de forma adecuada. De nuevo, se aplican sucesivos compactados hasta que las líneas no varían, porque de hacerlo se saltarían esos criterios. Llegados a ese punto, las líneas se estructuran en bloques de subtítulos que se encaminarán hacia el buffer de salida.

Un módulo importante es el **procesador de subtítulos**[10]. Dentro de este elemento funcional, una vez se han generado los subtítulos es posible que persista el problema de tiempos de referencia mal inferidos que se explicó en el apartado 4.6.2.2. Por esto, aún es necesario aplicar en algunas ocasiones una corrección adicional para adecuar los tiempos de referencia aún más. El procesador de subtítulos funciona en dos modos de operación principalmente:

- Cuando se dispone de unos subtítulos ya formados en el segundo flujo de entrada al sistema completo (por ejemplo si se dispone de subtítulos de rehablado proporcionados por una cadena de televisión que hay que sincronizar pero están ya formados, es decir, hay que corregir los tiempos pero no la

[10] En el diagrama de la arquitectura (figura 5), Processor.

estructura). En este caso se dispone de un dato adicional muy importante para utilizar a la hora de asignar los tiempos de referencia: la duración de los grupos de subtítulos. Por ello, si un subtítulo generado por la herramienta dura más o menos tiempo que el subtítulo original, es posible aplicar ciertas correcciones para ajustar las duraciones de forma óptima.

- En caso contrario (cualquier caso de diferido donde sólo se disponga del texto transcrito, o bien en directo por ejemplo). En esta situación, un subtítulo que dura medio segundo es ilegible (depende de la longitud, pero en general una línea dura en media unos dos segundos y por tanto un subtítulo dura de 3 a 4 segundos habitualmente). Por tanto se van procesando los subtítulos y si la duración es ínfima o bien muy grande, se corrige. En general, la pérdida de convergencia al alinear se traduce en una ráfaga de 3-4 subtítulos cortos y uno muy largo o bien al contrario. Es decir, si se alargan esos subtítulos cortos a posteriori, implícitamente se está repartiendo el tiempo del subtítulo de mayor duración entre ellos. El caso contrario es análogo, si un subtítulo es muy largo y se acorta, ese tiempo adicional se utiliza para alargar los subtítulos siguientes, que eran excesivamente cortos, reequilibrando el conjunto de esa manera. Esta corrección no implica en este segundo caso un sincronismo perfecto, ya que si no ha habido convergencia al alinear no es posible afirmar que existe buena sincronía, pero al menos permite la legibilidad de los subtítulos, lo cual es una gran ventaja (sin el postprocesado, estas regiones no solo no serían síncronas sino que además habría subtítulos ilegibles).

Una vez se tiene los Caption bien formados, se encolan en el buffer de salida, de modo que se puedan utilizar en según qué situaciones de la manera que se precise. Por ejemplo, en diferido se utiliza un último módulo que genera un archivo de subtítulos en formato .srt que puede ser reproducido en cualquier equipo convencional por cualquier usuario. En directo se utilizarán del modo más conveniente, en general se almacenarán para ser sincronizados posteriormente con el vídeo.

4.6.2.2 Interfaces

Al igual que se han diseñado las principales entidades funcionales del sistema, se han definido las interfaces entre los mismos. Esto es absolutamente necesario primero para comprender por completo el funcionamiento del sistema, ya que hay que conocer qué tipos de información se intercambian entre las distintas partes de la arquitectura, y segundo para poder determinar de qué forma se realiza dicho intercambio de datos.

Gracias al modelado de la información y estructura jerárquica presentados en el apartado 4.6.1, y teniendo en cuenta que el sistema implementado es una aplicación que funciona sobre varios hilos de ejecución paralelos, la definición de interfaces más sencilla y directa es mediante colas. Las colas son estructuras de datos de tipo FIFO (First In First Out), muy útiles en situaciones en las que el flujo de información es completamente lineal. Dado que en el caso que nos ocupa los elementos del sistema toman un flujo lineal de información de entrada, le aplican determinadas operaciones y

algoritmos y generan a partir de ello otro caudal lineal de información de salida, este tipo de estructuras es idónea. Obviamente se precisa también de sincronía en el acceso a las colas, ya que al trabajar sobre distintos hilos de ejecución la velocidad de cada uno de los flujos que transcurren por el sistema no es necesariamente la misma y por ello podrían surgir problemas en caso de tratar de acceder a una de las colas desde dos hilos de ejecución distintos para extraer información.

Teniendo en cuenta estas premisas, se han implementado unas estructuras de tipo FIFO que además tengan en cuenta la sincronía en el acceso a las mismas. Éstas, que se muestran en la figura 4.1 como Queue, son utilizadas en todas las interfaces portando distintos tipos de información en cada caso.

Como se puede apreciar en la figura, al reconocedor de voz le llega la señal de audio a procesar. Dicha señal es transcrita a texto y ese texto se encapsula en objetos de tipo Word que son encolados en una de las dos entradas del motor de alineamiento. La segunda entrada también posee una interfaz similar, esto es, consiste en otra cola de objetos Word que provienen, según el caso, o bien de algún tipo de archivo o bien del ASR que transcribe el audio del rehablador. La siguiente interfaz definida se encuentra entre el motor de alineamiento y el generador de subtítulos, y debido a la naturaleza de las operaciones aplicadas de alineamiento e inferencia de tiempos, es también una cola de objetos Word. Como se ha mencionado anteriormente, el generador de subtítulos toma las palabras y las transforma en bloques de subtítulos, por lo que la salida de dicha entidad no es más que una cola de objetos Caption, listos para ser escritos en un archivo de subtítulos o para ser utilizados de cualquier otro modo.

4.6.3 Arquitectura modo diferido

Una vez explicada la estructura y el funcionamiento de la parte común del sistema a ambas arquitecturas, se presentan las peculiaridades de cada una de ellas.

Como se ha explicado a lo largo de este documento, el modo diferido u offline hace referencia al modo de funcionamiento en el que se dispone de texto en cualquier tipo de formato (por ejemplo texto plano, subtítulos en formato .xml, .srt, etc.) y a partir del mismo se construye una versión sincronizada de los subtítulos de algún vídeo o emisión que no estén en directo. Lo que se espera conseguir, ya que el vídeo a subtitular no es en directo, es un archivo de subtítulos que poder utilizar tras completarse la ejecución de la herramienta.

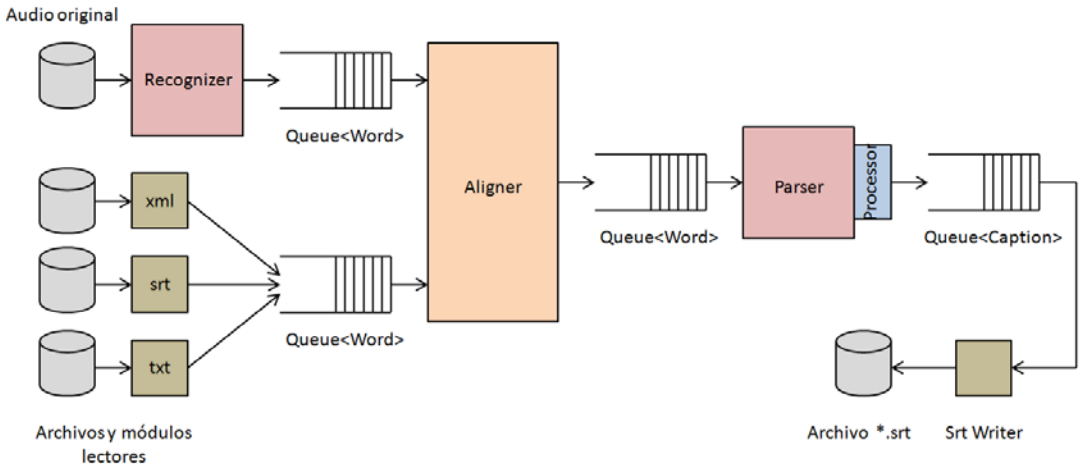


Figura 6: Arquitectura del sistema en modo diferido

El funcionamiento en este caso es como ya se ha comentado en toda la parte que es común. En cuanto a los elementos particulares del sistema, como se observa en la figura anterior se ha implementado una librería que ofrece una serie de módulos que sirven para leer y escribir en diferentes tipos de archivos[11]. Por ejemplo, se pueden leer archivos en formato .txt y .srt, de forma que se extraen las palabras del archivo (el texto en sí), la estructura de dichas palabras si procede (por ejemplo en un .srt ya hay bloques de subtítulos escritos con una organización establecida, por lo que se guarda en los objetos Word extraídos referencias para conocer la estructura que mantenían –a qué línea pertenecían y a qué bloque de subtítulos pertenecía esa línea-) e información relevante de duración en su caso (del mismo modo, en un .srt aparece la duración de cada bloque de subtítulos, información que se mantiene para una posterior corrección en caso de errores al inferir tiempos). Así mismo, se aprecia que a la salida del sistema hay un módulo complementario a los mencionados en el párrafo anterior, que sirve para realizar el proceso inverso por ello, es decir, toma una serie de subtítulos y los escribe en un archivo para que el usuario final pueda reproducirlo cuando sea necesario.

Una consideración importante es que en este caso se tiene más libertad en cuanto a la elección de modelo del lenguaje a utilizar durante el reconocimiento, esto es, dado que se dispone del vídeo a priori, es posible obtener en muchas ocasiones el modelo de la persona que habla, mejorando notablemente la calidad de la sincronización obtenida.

4.6.4 Arquitectura modo directo

El modo de funcionamiento en directo u online implica que los datos de partida a partir de los cuales se logrará obtener una versión de subtítulos sincronizada provienen de un rehablador. El rehablador se dedica a escuchar cuanto se dice durante la emisión y a repetirlo de una forma compacta (dado que no es posible que se repitan todas las

[11] En la figura de arquitectura en modo diferido (6), los módulos lectores (en verde) son los que permiten extraer el texto que se introduce en la segunda entrada del sistema.

palabras emitidas, la persona que rehabla debe no sólo resumir, sino interpretar lo que se dice para ser capaz de retransmitirlo de una manera concisa pero sin olvidarse de ningún detalle relevante). Tras la ejecución de la herramienta se pretende obtener exactamente lo mismo que en el caso anterior, una versión de subtítulos sincronizada con la emisión. En este caso, la sincronización entre subtítulos y el vídeo con el audio requiere retrasar la proyección de la imagen de alguna forma (por ejemplo bufferizando en el receptor TDT del usuario la información y manteniéndola un breve tiempo -20 segundos es un valor típico-), y de esa forma es factible ofrecer una versión de los subtítulos que genera el rehablador con unos tiempos de presentación relativos a la emisión completamente adecuados. Otra posibilidad es retrasar la emisión del programa un cierto tiempo (que será equivalente al mencionado antes, es decir, típicamente 20 segundos) para permitir la emisión de los subtítulos en el instante correcto.

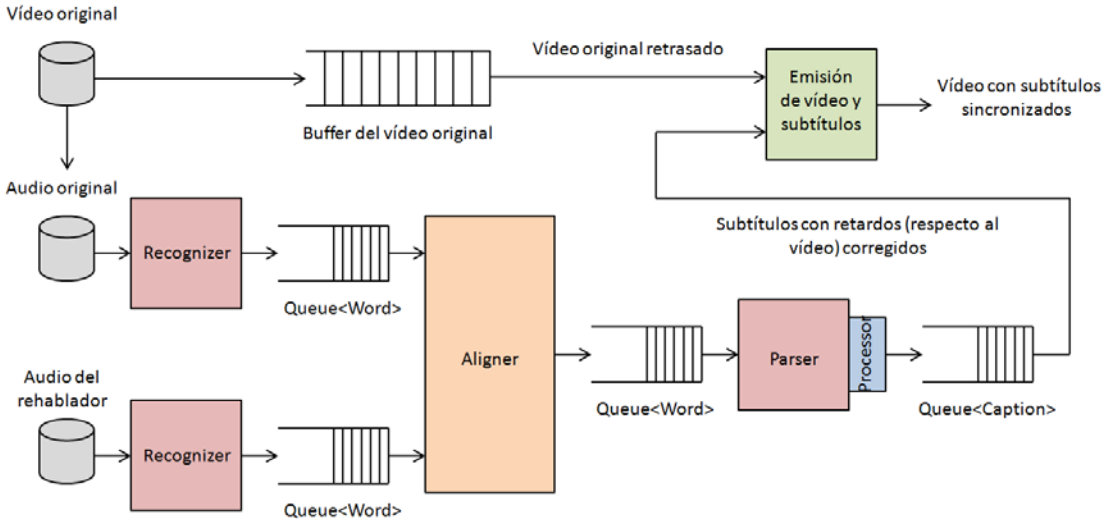


Figura 7: Arquitectura del sistema en modo directo

Como se puede observar, la diferencia respecto al caso anterior es de dónde proviene el segundo flujo de entrada al motor de alineamiento. En este caso no se dispone de ninguna clase de transcripción que se pueda cargar en el sistema de antemano, sino que las palabras del flujo son generadas por el rehablador en tiempo real. Lo que este individuo pronuncia ha sido procesado por un segundo módulo de reconocimiento de habla, adecuado a su perfil acústico y modelo del lenguaje para que la transcripción sea lo más fidedigna posible. Recuérdese que estas ráfagas de palabras son las que se utilizan para alinear debido a que su texto es en teoría correcto, mientras que las provenientes del primer ASR son para obtener los tiempos de referencia adecuados. Teniendo esto en consideración, y asumiendo que el rehablador no sólo resume lo que se dice, sino que interpreta (modificando el texto real que se dice) y además se puede equivocar, el alineamiento es algo más complicado y es más probable perder la convergencia. Por otro lado, los subtítulos obtenidos al final del proceso, aunque no suponen una representación ideal del audio del programa, ya que el texto que los

conforma proviene del rehablador, que ha comprimido, interpretado y errado lo que se dice, suponen una mejora significativa de la calidad actual del subtitulado en directo por estar sincronizados. La otra gran diferencia es que el vídeo se debe retardar unos segundos (en emisión o en recepción) para después emitir en el destino el vídeo con los subtítulos sincronizados (eso está fuera del ámbito del proyecto, que proporciona lo que en la figura se muestra como "*subtítulos con retardos corregidos"*).

En definitiva, hay que tener siempre presente que los subtítulos que se obtienen no son perfectos en cuanto a la literalidad con la cual plasman lo que se comenta durante la emisión, pero eso no es lo que se pretende. Lo que se busca en realidad es corregir todos los retardos que suceden durante dicha emisión, corrección que se aplica de manera individual porque todos son retardos temporales variables y no es posible tratar todos desde un punto de vista de conjunto, además de que en tiempo real la única opción es ir corrigiendo según se dispone de datos paulatinamente (no es posible esperar para obtener algún tipo de regla o generalizar).

En este caso, dado que a priori se desconoce la naturaleza del audio original que se va a utilizar para comparar (por ejemplo, no se dispone de modelos acústicos del presentador de un programa determinado que utilizar), normalmente se utilizará un perfil genérico, sin entrenar del ASR. No obstante, dependerá de cada caso el uso que se le dé al sistema.

Tras realizar todas las operaciones intermedias y obtener subtítulos sincronizados, el último paso es almacenarlos en memoria para ser posteriormente sincronizados con el flujo de vídeo retardado.

5. Pruebas y resultados

5.1 Introducción

Una vez cubiertas las etapas de desarrollo del Trabajo de Fin de Grado, se detallan las pruebas efectuadas y los resultados obtenidos en las mismas. El objetivo de las pruebas es, principalmente, el de constatar que el sistema es capaz de, a partir de texto y audio, obtener subtítulos sincronizados. Evidentemente esto no es trivial, e implica en sí mismo comprobar si el sistema puede alinear secuencias adecuadamente (esto es, si el algoritmo diseñado e implementado funciona y, por tanto, es utilizable en proyectos de mayor envergadura). Asimismo, también permite comprobar que el algoritmo aplicado para la inferencia de tiempos de palabra es válido, y por ello permite solucionar de una manera automática y adaptativa los retardos variables asociados a la generación de subtítulos en escenarios de rehablado. En tercer lugar, posibilita determinar si los subtítulos que se generan son adecuados en cuanto a forma, tamaño y particionado.

Por otro lado, el conjunto de pruebas realizadas permite verificar dos factores importantes. Por un lado, si el sistema es viable en sí, ya que las limitaciones técnicas y algorítmicas expuestas anteriormente exigen un ajuste de los umbrales de decisión para el alineamiento extremadamente preciso, que permita maximizar la probabilidad de alineamiento correcto teniendo en cuenta los posibles errores tanto por los ASR como por los algoritmos. Por otro, determinar bajo qué condiciones el sistema es viable. En otras palabras, en qué situaciones es posible utilizar estos algoritmos y funcionalidades con garantías de éxito.

A continuación se presentan las pruebas realizadas durante y después de la implementación de las funcionalidades requeridas por el proyecto.

5.2 Pruebas y resultados

5.2.1 Pruebas de alineamiento

El núcleo del sistema es el algoritmo de alineamiento, cuya validez ha quedado demostrada durante la fase de pruebas. El algoritmo de alineamiento es la parte más importante y por tanto debe funcionar de manera adecuada para asegurar que, en fases posteriores, los posibles problemas no derivan de un mal alineamiento. Para ello se implementó el algoritmo de Needleman-Wunsch, de programación dinámica, explicado en el apartado de diseño, y se procedió a su adaptación para el alineamiento de secuencias de palabras, para evaluar sus prestaciones. En este punto, las pruebas se han obtenido mediante la consola de comandos de Windows y así se expondrán las imágenes pertinentes. Tomemos dos conjuntos de palabras como referencia y evaluemos los resultados obtenidos. Por ejemplo, utilizaremos las secuencias *"Hola, me llamo Alejandro y redacto un apartado"* y *"Hola me llamas redacto un apartado"*, para irlas modificando y apreciando los resultados. Lógicamente la primera será la secuencia de referencia y la segunda una secuencia con errores que podría provenir de un ASR. La idea es determinar hasta qué punto el alineamiento funciona, para poder discernir si es

posible alinear aplicando este algoritmo en este tipo de entornos. Si introducimos esas dos secuencias al algoritmo, el resultado que obtenemos es el siguiente.

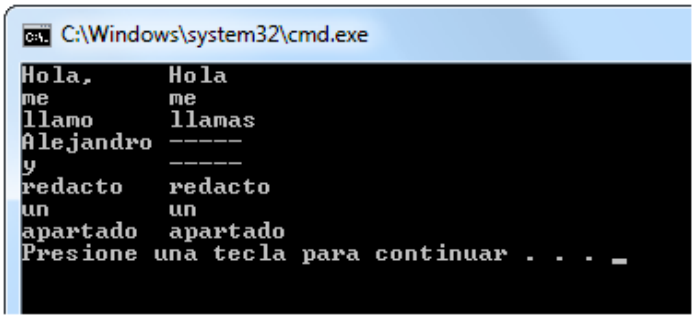


Figura 8: Alineamiento 1

Se aprecia claramente que el alineamiento es bueno. Muchas palabras eran exactas, por lo que el alineamiento es adecuado. Además, en la segunda secuencia faltaban dos palabras, que aparecen marcadas tras alinear. Es justo la respuesta que se esperaba. Igualmente, se ha alineado *llamo* con *llamas*, puesto que comparten la misma raíz.

Supongamos que ahora los conjuntos de palabras son más escuetos. El alineamiento será más complicado. Si ahora suprimimos *redacto* del segundo conjunto, y *Hola*, *apartado* del primero y alineamos, obtenemos:

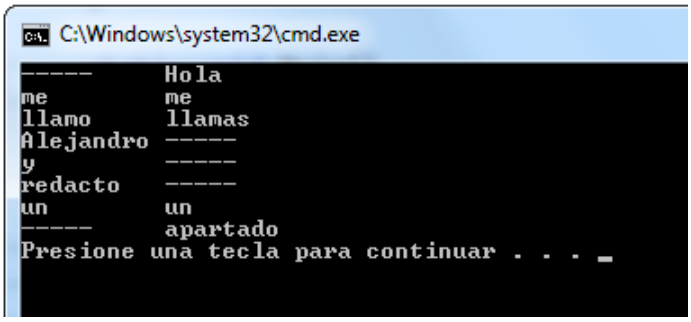


Figura 9: Alineamiento 2

Se ve que el alineamiento es exitoso de nuevo, apareciendo las palabras alineadas donde deben y los huecos en cada una de las secuencias donde no se ha encontrado que haya dos palabras que correspondan.

Ahora se expone un caso menos exitoso, que permite comprobar la dificultad del alineamiento. Supongamos que suprimimos *llamo* de la primera secuencia. Se obtiene:

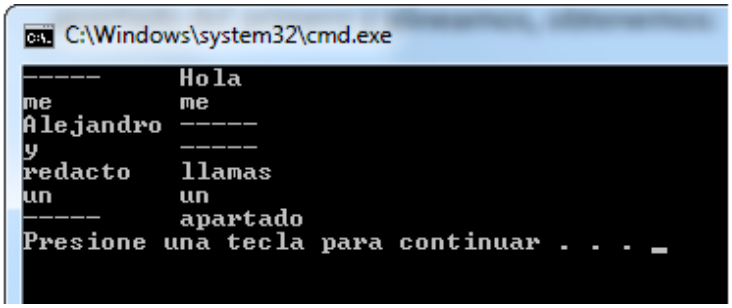


Figura 10: Alineamiento 3

El alineamiento no es del todo correcto. Funciona bien, pero alinea *redacto* con *llamas*. Esto no es un error del algoritmo, sino que durante el proceso de alineamiento se ha determinado que alinear esas dos palabras es lo más probable (se trata de una sustitución, no como en los casos anteriores en los que solo había inserciones o borrados). En puntos de implementación más avanzados se contempla este problema: hay casos en los cuales una sustitución es adecuada, ya que lo que interesa son las marcas de tiempo de esa palabra y no el contenido en sí, por lo que si el ASR proporciona tres palabras y se han alineado la 1 y la 3, la segunda no importa que no coincida, sus tiempos son los que se necesita. En cambio en otros casos un mal alineamiento puede generar errores al inferir tiempo. En la figura anterior se ve que la sustitución implica que, como *me* y *llamas* (columna derecha) son palabras adyacentes, los tiempos de *llamas* no son válidos para inferir los de la columna de la izquierda, porque van a inducir a error. En ese caso se utilizará para asignar los tiempos las palabras *me* y *un*, alineadas correctamente.

En definitiva, el alineamiento es un proceso complicado con muchas implicaciones, pero en la primera fase de implementación el conseguir unos resultados como los anteriores permite dar por demostrada la viabilidad del algoritmo, ya que el alineamiento adaptado a palabras es en general bueno y, por tanto, es posible utilizarlo a niveles más complejos.

5.2.2 Pruebas de inferencia de tiempos

Una vez comprobado que el alineamiento es útil y que el nivel de calidad es adecuado, se pasa de trabajar alineando palabras (cadenas de caracteres) y se implementa el modelo básico de datos expuesto en el capítulo 4.6.1. Por consiguiente, el siguiente paso es el de asignar tiempos cuando hay coincidencia en el alineamiento o inferirlos adecuadamente en caso contrario, trabajando con objetos Word que constan de una palabra (lo que se alinea) y las marcas de tiempo de inicio y tiempo de fin. Las pruebas de inferencia de tiempos son muy simples, ya que dependen directamente de las de alineamiento. Dicho de otro modo, al alinear una secuencia de referencia con una secuencia con tiempos (y errores en el texto), simplemente a las palabras alineadas de la primera se le copian los tiempos de sus correspondientes parejas en la segunda. La visualización de las pruebas es análoga al apartado anterior, simplemente utilizando unas secuencias de test y comprobando que se han copiado los tiempos adecuados. Tras la asignación de tiempos, se infieren los no asignados según un sencillo algoritmo que obtiene los tiempos de letra y calcula lo que dura cada palabra. Los resultados se pueden dividir en dos subconjuntos: cuando el alineamiento ha funcionado bien, se observa que los tiempos calculados son correctos, mientras que si se ha perdido la convergencia, hay palabras cuya duración es muy pequeña o excesivamente grande. En esos casos se vuelve a ajustar la duración aplicando un postprocesado como se explica en el capítulo de diseño.

5.2.3 Pruebas de generación de subtítulos

Si el alineamiento y el cálculo de tiempos son adecuados, se puede disponer de texto con marcas de tiempo listo para generar subtítulos. Por ello, en un tercer conjunto de pruebas es relevante comprobar si el Parser de subtítulos los genera de una forma correcta, es decir, si se cumplen las especificaciones de número máximo de caracteres por línea y correcto particionado de los subtítulos, para almacenarlos y poderlos revisar posteriormente. Tanto estas pruebas como las de inferencia de tiempos se plasmarán en imágenes en el siguiente apartado, porque de otra forma sería complicado explicarlas de una manera precisa en esta sección. Cualitativamente, se puede mencionar que el cumplir que el número de caracteres no excede los 37 en una línea es muy sencillo de implementar, mas el particionado es relativamente complejo. Se han generado listas de palabras clave (tales como conjunciones o preposiciones), que junto con los signos de puntuación y la restricción anterior permiten particionar de un modo relativamente correcto. Por otro lado, los subtítulos son generados sin determinar otros aspectos como el color, la caja, etc. Ese no es el objetivo del proyecto, sino que la obtención de subtítulos de una forma ágil y cómoda supone un valioso añadido que permite facilitar la labor de subtitulado en diferido. Un ejemplo de un fragmento de archivo de subtítulos es el siguiente. Así es como se parten los subtítulos, de manera automática:

```
39
00:02:15,450 --> 00:02:18,960
economía y empleo, el primero;
política social el segundo

40
00:02:18,961 --> 00:02:21,591
y un tercero más amplio y variado
en el que podemos hablar

41
00:02:21,592 --> 00:02:25,130
de calidad democrática, la posición
de España en el mundo

42
00:02:25,131 --> 00:02:27,577
y la política en general.

43
00:02:27,578 --> 00:02:30,990
El debate es suyo y ustedes son
los principales protagonistas.

44
00:02:30,991 --> 00:02:36,191
Este es el formato nítido,
claro, del cara a cara.
```

El primer número indica el número de subtítulo, el identificador.

La siguiente línea muestra el tiempo de inicio y el tiempo de fin de cada bloque de subtítulos, mostrado con un formato hh:mm:ss,mmm (horas, minutos, segundos y milisegundos).

Después aparece el par de líneas que conforman un subtítulo. Es lo que aparecerá en pantalla, durante el tiempo dictaminado por la línea anterior.

Figura 11: Ejemplo de parseo de subtítulos

No obstante el proyecto gira en torno al alineamiento y el cálculo de tiempos, algo de lo que se apreciará su valor en el siguiente epígrafe.

5.2.4 Pruebas del sistema completo

Como se ha mencionado, el sistema está diseñado para operar en dos entornos específicos bien diferenciados: diferido y directo. Además, la arquitectura y funcionamiento son análogos en ambos casos, la principal diferencia es que el texto de referencia en el primer escenario se obtiene a partir de un archivo y en el segundo a partir de un reconocedor de habla utilizado por un rehablador. Debido a que no se dispone en el CESyA de un profesional del rehablado para realizar las pruebas, se ha tratado de emular esta situación para comprobar si el sistema completo funciona.

Este conjunto de pruebas consisten en, dados un audio perteneciente a un vídeo en diferido, y la transcripción textual del mismo, generar subtítulos en formato .srt ejecutando la herramienta y visualizar los resultados. Evidentemente esto probaría la eficacia del sistema en diferido, por lo que emular el directo se realiza mediante dos métodos: primero, la degradación de la transcripción de referencia. Lógicamente, el ASR del rehablador comete errores y genera un texto no ideal, por lo que si se simula ese problema, es posible comprobar la viabilidad del sistema en ese caso. Mencionar que para algunas pruebas se han conseguido los subtítulos reales de rehablado, proporcionados por RTVE al CESyA. Por otro lado, implícitamente en el diseño conjunto la forma en que se extraen de las colas de entrada las secuencias de palabras simula tráfico a ráfagas, es decir, pese a disponer de la transcripción completa (y, por tanto, poder disponer de un conjunto de miles de palabras de partida), se extraen las palabras a bloques, simulando una situación como la del rehablado, en la cual el reconocedor va soltando listas de palabras cada poco tiempo. Por ello, pese a ser pruebas en diferido se puede afirmar que funcionarían en el escenario de rehablado y, por consiguiente, podrían extrapolarse al caso de directo aplicando los ajustes oportunos.

Para efectuar este conjunto de pruebas se ha escogido una serie de vídeos representativos que se adecuaran perfectamente a los criterios establecidos, es decir, entornos controlados, con audio limpio y libre de errores y/o perturbaciones. La selección de vídeos utilizados se compone en parte de grabaciones realizadas por profesores de la propia Universidad Carlos III de Madrid impartiendo clase de un máster a distancia en materia de accesibilidad, y por otro lado por algunos vídeos obtenidos de emisiones de RTVE, en particular de los programas 59 segundos y Las mañanas de la 1. En los primeros se experimenta en entornos sin distorsiones, con modelos acústicos específicos de cada profesor en cuestión primero (ya que solo hay un interlocutor) y luego modelos sin entrenar, mientras que en los segundos se trata de generar los subtítulos utilizando un perfil genérico debido a que son varias las personas que intervienen. De esta manera es posible evaluar el impacto que tiene la elección del modelo en el ASR en la calidad del alineamiento y de la sincronización de los

subtítulos. Además, como las transcripciones utilizadas para ir alineando (el texto de referencia) se obtienen manualmente, degradándolas se puede estudiar de qué manera influye en el alineamiento y por tanto en la sincronización. La duración de las muestras escogidas es variable, desde dos minutos hasta veinte en el caso máximo. De todos modos el tiempo que se tarda en efectuar todo el proceso de alineamiento, inferencia de tiempos y generación de subtítulos es aproximadamente de un 20% del tiempo de la muestra. Es decir, a partir de las muestras de 20 minutos se obtiene el archivo con los subtítulos sincronizados respecto al vídeo en aproximadamente 4 minutos. Este tiempo de procesado es ligeramente variable y aumenta si se pierde la convergencia, porque hay que alinear secuencias más largas en este caso (y la complejidad del algoritmo aumenta).

Las pruebas para el conjunto de muestras se estructuran en dos niveles. El primero y más inmediato, el de medir la sincronización de las frases con alto grado de parecido entre la transcripción y el vídeo. El segundo implica diseñar un modelo de pruebas que permita particularizar a las situaciones en las cuales el índice de parecido sea bajo, y determinar en esos casos cuán sincronizadas se encuentran las frases. Este caso es bastante más complejo que el anterior y está fuera del ámbito de este proyecto. Si es posible llegar a ese nivel de concreción, sería un gran paso de cara a corregir las faltas de sincronismo en cualquier situación posible.

Para cada muestra, por tanto, se ha obtenido su transcripción (literal inicialmente) de manera manual y se ha utilizado el sistema para generar los subtítulos sincronizados. De estos subtítulos se ha medido el retardo respecto al vídeo manualmente utilizando SoundForge[12]. Asimismo se ha comparado meticulosamente el texto de referencia con la transcripción del audio obtenida por el ASR, para determinar en qué instantes la calidad del reconocimiento no era adecuada y evaluar los posibles motivos (por ejemplo debido a ruidos ambientales).

Una de las pruebas realizadas más importantes es la del debate político ofrecido por TVE entre los entonces aspirantes a presidente del Gobierno Mariano Rajoy y Alfredo Pérez Rubalcaba, anterior a las pasadas elecciones. Ese vídeo es determinante por dos motivos principalmente. Primero, porque se realizan las pruebas con los subtítulos de rehablado generados en los estudios de RTVE, obteniéndose unos buenos resultados dadas las diferencias entre la transcripción literal del debate y el propio rehablado[13]. Segundo, porque en el vídeo intervienen tres interlocutores: Mariano Rajoy, Alfredo Pérez Rubalcaba y el moderador, Manuel Campo Vidal. En este caso, utilizando un modelo genérico sin entrenar del reconocedor de habla, el alineamiento es bastante bueno, aunque se mostrará lo que de verdad genera el ASR en contraposición a lo que

[12] SoundForge es un software de edición de audio que implementa herramientas para grabar, modificar, medir y realizar cualquier tipo de operación sobre señales de audio.

[13] Para visualizar la demostración completa, en la cual aparecen tanto los subtítulos con retardo generados mediante rehablador por RTVE como los subtítulos sincronizados respecto al vídeo obtenidos utilizando la herramienta, el lector puede acceder al siguiente enlace:

1. Demo Debate político 2011: http://www.youtube.com/watch?v=1NaR6tDKrrE

dicen los interlocutores para poder apreciar el alcance de los errores que produce. Utilizaremos ese ejemplo para explicar este conjunto de pruebas.

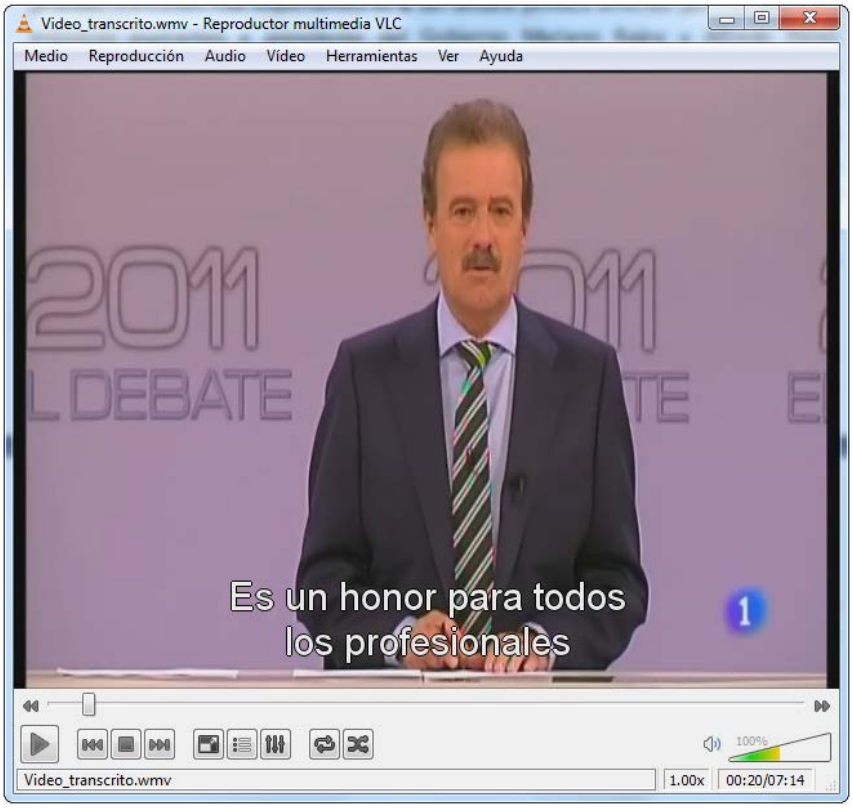


Figura 12: Demo debate 2011 seg. 20

El subtítulo que se muestra en la imagen es el generado mediante alineamiento y sincronización. El texto es parte de los subtítulos proporcionados por TVE, generados por un rehablador y sincronizados por la herramienta desarrollada, tras aplicar el reconocimiento de habla al audio original. Lo primero que hay que destacar es la diferencia entre el texto de referencia y el generado por el ASR a partir del audio. En este caso concreto, la diferencia es la siguiente en los primeros 25 segundos de vídeo:

Texto del rehablador	Texto originado por el ASR
"Buenas noches España, buenas noches Europa. En nombre del Academia de las Ciencias y las Artes de Televisión, les damos la bienvenida a este gran debate. Es un honor para todos los profesionales y para todos los profesionales que integran la academia haber tenido la confianza de los dos partidos".	*"Sánchez por las noches español por las noches uno paz buenas tardes América el nombre de academia de las ciencias y las artes de televisión de una la bienvenida a este gran debate de dos mil es un humor para todos los profesionales y por las televisiones que integran la academia haber tenido la confianza de los dos partidos".*

La cantidad de errores es notable, y aun así el resultado del reconocedor es en general más que aceptable teniendo en cuenta el perfil sin entrenar. En este mismo debate, otros fragmentos ofrecen estos resultados en la transcripción:

Texto del rehablador	Texto originado por el ASR
"que requiere nuevas medidas. Si soy elegido presidente del gobierno, buscaré un acuerdo para el empleo. Es una gran causa nacional". (min 6:29-6:40)	*"en duda exige nuevas técnicas yo desde CD-ROM endometrio comprometan sus tres aprestos primero a su sol y elegido presidente gobierno buscan un acuerdo para el empleo es una gran causa nacional"*.
"en España. Alfredo Pérez Rubalcaba, candidato del PSOE. Mariano Rajoy Brey, candidato del Partido Popular". (min 1:43-1:50)	*"los que tiene mayor presentación proletaria hace Lope Rubalcaba candidato Partido Socialista con coches albaneses bajo el candidato del Partido Popular"*.

A pesar de las más que notables diferencias, ambos pares de secuencias se van alineando exitosamente en general al ejecutar la herramienta. En la siguiente imagen se muestra el retardo que se puede corregir al utilizar este sistema.

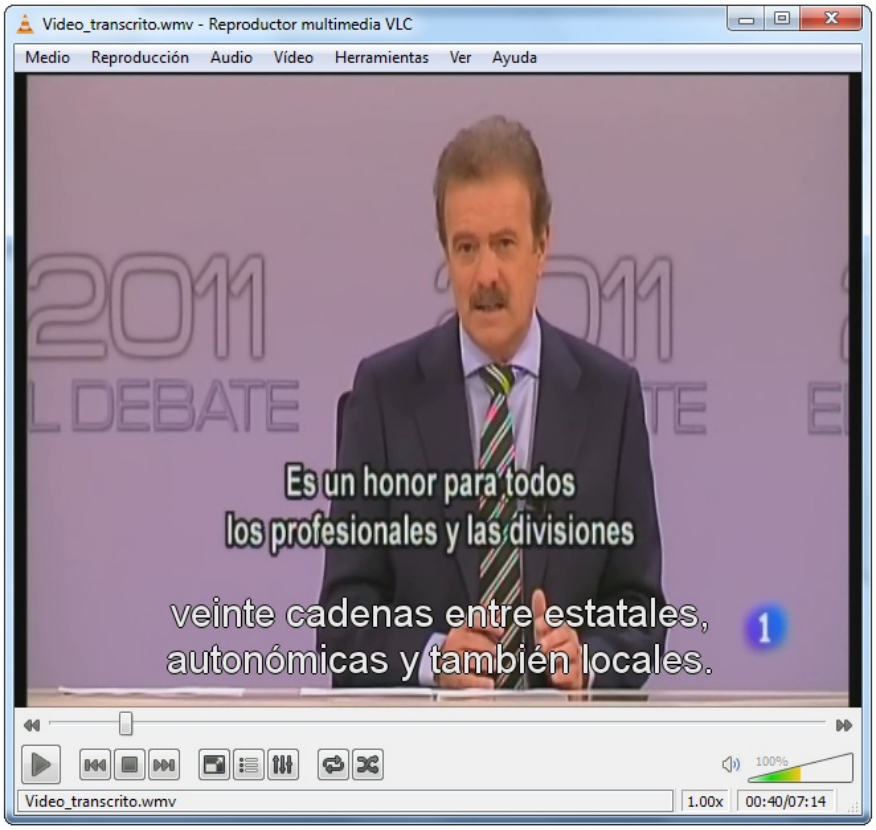


Figura 13: Demo debate 2011 seg. 40

Los subtítulos que aparecen en el centro de la pantalla son los que se mostraron en la emisión en directo del evento. Se puede comprobar que aparecen en pantalla en el segundo 40 aproximadamente. Pero lo que aparece se mencionó en el segundo 20, con lo cual es prácticamente imposible que un discapacitado auditivo sea capaz de seguir la trama con esos subtítulos. Debajo, continúan apareciendo los subtítulos sincronizados generados por el sistema desarrollado, correspondientes al instante preciso en el cual el señor Campo Vidal dice esas palabras. Las palabras de Campo Vidal en los subtítulos de rehablado de RTVE aparecerán en pantalla unos 15 segundos más tarde (en la emisión original).

Hay casos en los que, al perder la convergencia al alinear, la sincronización no es idónea. No obstante esto se recupera automáticamente, volviéndose a presentar los subtítulos de forma sincronizada tras unos segundos. Esta es una de las mejoras futuras, optimizar la calidad del postprocesado para adecuar la duración de los subtítulos mal generados debido a errores de alineamiento. A pesar de ello se observa que es posible corregir el retardo mediante el alineamiento y el cálculo de tiempos de palabra. Con las soluciones actuales un discapacitado es, en muchas ocasiones, incapaz de seguir la acción. De hecho este ejemplo es muy interesante porque mientras habla Rubalcaba aparecen subtítulos correspondientes a Mariano Rajoy, lo cual es completamente desastroso. Aplicando los algoritmos generados es posible solventar en gran medida ese problema, por lo que es una solución interesante sobre la que merece la pena investigar.

6. Plan de proyecto

6.1 Introducción

La planificación surge de la necesidad de estimar los costes económicos y temporales que requiere el desarrollo de un proyecto, así como del control de los factores que pueden alterar su evolución. De esta manera, aplicando una metodología de trabajo definida, es posible que cada nuevo proyecto se construya en base a la experiencia acumulada de los anteriores. Además, permite identificar a tiempo las causas de los problemas, y por ello corregirlos de manera eficaz.

En esta sección se va a realizar una estimación de los recursos empleados a lo largo del desarrollo del proyecto, desde un punto de vista temporal, humano, material y económico.

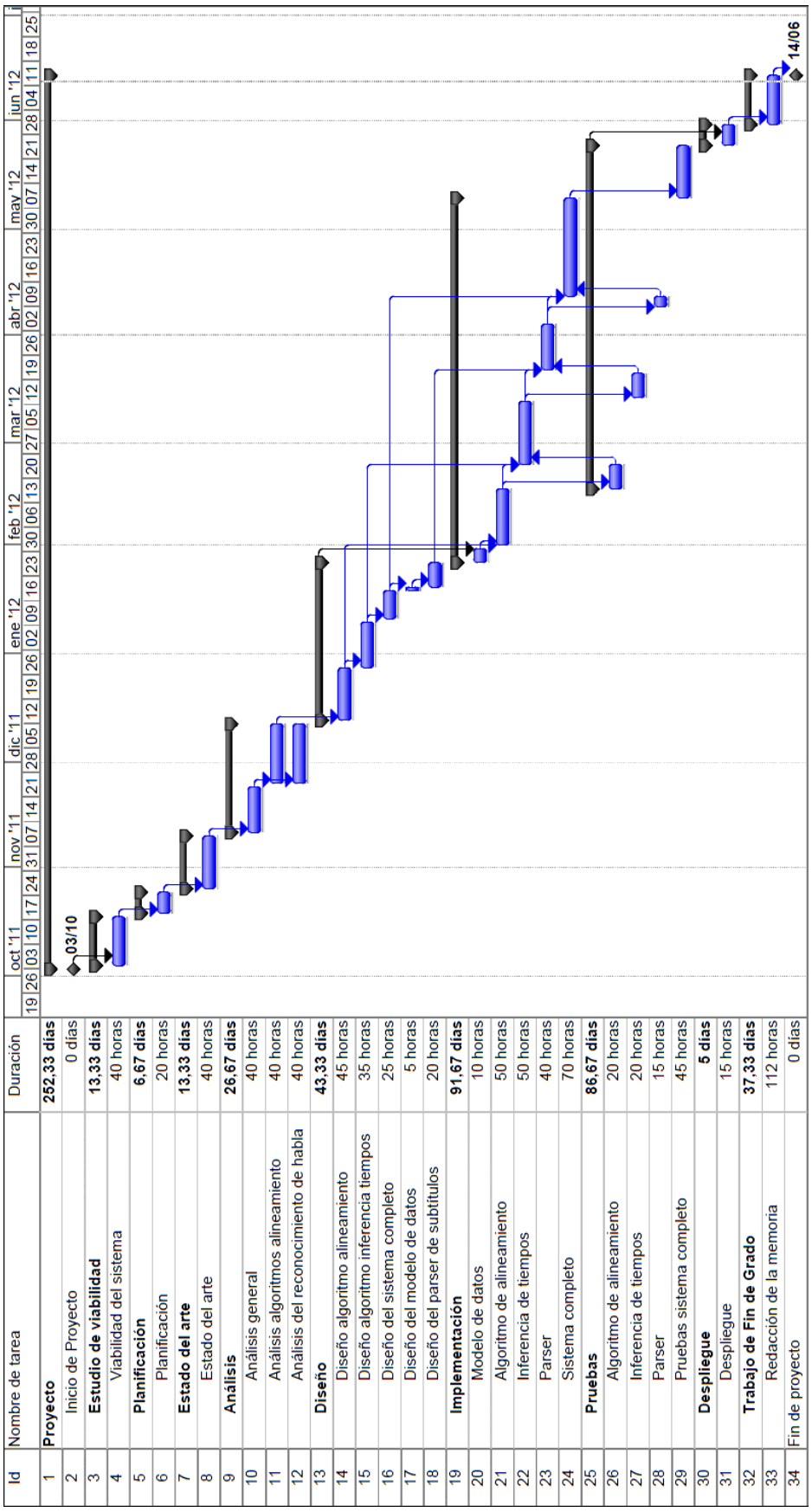
6.2 Estimación de recursos temporales

La fecha de inicio del proyecto se establece el día 3 de octubre de 2011. En base a la planificación, la fecha de finalización se sitúa el día 14 de junio de 2012. Entre ambas fechas se sitúa el desarrollo del proyecto, con un total de 797 horas. El reparto de horas queda establecido en la siguiente tabla:

Tabla 3: Recursos temporales por fases del proyecto

Tarea		Horas
1	Estudio de viabilidad	40
2	Planificación	20
3	Estado del arte	40
4	Análisis	120
5	Diseño	130
6	Implementación	220
7	Pruebas	100
8	Despliegue	15
9	Redacción del proyecto	112
Horas totales		797

Por otro lado, el diagrama de Gantt que se expone en la siguiente figura muestra, de una forma ordenada, la sucesión de cada una de las tareas que componen el desarrollo del proyecto.



Id	Nombre de tarea	Duración
1	**Proyecto**	**252,33 días**
2	Inicio de Proyecto	0 días
3	**Estudio de viabilidad**	**13,33 días**
4	Viabilidad del sistema	40 horas
5	**Planificación**	**6,67 días**
6	Planificación	20 horas
7	**Estado del arte**	**13,33 días**
8	Estado del arte	40 horas
9	**Análisis**	**26,67 días**
10	Análisis general	40 horas
11	Análisis algoritmos alineamiento	40 horas
12	Análisis del reconocimiento de habla	40 horas
13	**Diseño**	**43,33 días**
14	Diseño algoritmo alineamiento	45 horas
15	Diseño algoritmo inferencia tiempos	35 horas
16	Diseño del sistema completo	25 horas
17	Diseño del modelo de datos	5 horas
18	Diseño del parser de subtítulos	20 horas
19	**Implementación**	**91,67 días**
20	Modelo de datos	10 horas
21	Algoritmo de alineamiento	50 horas
22	Inferencia de tiempos	50 horas
23	Parser	40 horas
24	Sistema completo	70 horas
25	**Pruebas**	**86,67 días**
26	Algoritmo de alineamiento	20 horas
27	Inferencia de tiempos	20 horas
28	Parser	15 horas
29	Pruebas sistema completo	45 horas
30	**Despliegue**	**5 días**
31	Despliegue	15 horas
32	**Trabajo de Fin de Grado**	**37,33 días**
33	Redacción de la memoria	112 horas
34	Fin de proyecto	0 días

Figura 14: Diagrama de Gantt

6.3 Estimación de los recursos económicos

A continuación se evalúan los costes económicos tanto materiales como humanos asociados a la ejecución del proyecto. En este proyecto se establece un plazo de amortización de 8 años para los elementos de hardware y 6 para el software empleado. En base a esto, los costes materiales asociados a la ejecución del proyecto se describen en la tabla siguiente:

Tabla 4: Recursos materiales del proyecto

Concepto	Precio	Coeficiente	Coste
PC sobremesa HP	900.00€	8.33%	75.00€
Microsoft Office 2007	219.27€	11.11%	24.36€
Microsoft Project 2007	149.00€	11.11%	16.55€
Microsoft Visio 2007	221.00€	11.11%	24.55€
Microsoft Visual Studio 2010	6990.00€	11.11%	776.66€
Dragon Naturally Speaking 11	99.50€	11.11%	11.05€
		Total	928.17€

Por otro lado, se contabilizan los recursos humanos utilizados a lo largo del trabajo desarrollado. En la siguiente tabla se muestran los perfiles de los profesionales requeridos para la realización de cada una de las tareas y el coste de los mismos, en función de sus honorarios y del número de horas empleado.

Tabla 5: Recursos humanos del proyecto

Perfil	Honorarios (€/hora)	Horas	Coste
Director de proyecto	113.00	10	1130€
Jefe de proyecto	75.00	50	3750€
Analista de servicios	63.00	195	12285€
Desarrollador de aplicaciones	50.00	320	16000€
Probador de software	25.00	100	2500€
		Total	35665€

Por último, se recogen los costes totales asociados a la ejecución del proyecto, es decir, el conjunto formado por los costes materiales y humanos que implicarían el desarrollo del proyecto.

Tabla 6: Costes totales del proyecto

Concepto	Valor
Recursos materiales	928.17€
Recursos humanos	35665€
Gastos (20% de RRHH)	7133€
Subtotal	43726.17€
Beneficios empresariales (20% de Subtotal)	8745.23€
Base imponible	52471.40€
I.V.A. (18% de base imponible)	9444.85€
Total	61916.25€

6.4 Tareas del proyecto

Las tareas del proyecto se han estructurado en una serie de etapas, de forma que abordar la ejecución del mismo fuera algo directo y planificado. Entre las etapas podemos distinguir:

- **Estudio de viabilidad del sistema:** A lo largo de esta etapa del proyecto se analizan las necesidades existentes y se estudia la situación. Se trata de comprender la problemática asociada al proyecto, estudiando los datos disponibles y realizando pruebas preliminares, para tratar de determinar primero si el proyecto es viable y, en caso de serlo, identificar elementos importantes que posteriormente ayudarán al desarrollo del mismo, así como los posibles escenarios de actuación.

  Entre otros elementos, se estudian las posibles restricciones existentes, como por ejemplo los umbrales de calidad de audio admisibles para que el sistema sea viable, bajo qué condiciones funcionan correctamente los ASR o mediante qué técnicas se pueden identificar palabras en el audio original de una emisión para después efectuar la sincronización.
- **Planificación:** En esta etapa se abordan los costes temporales y económicos del proyecto, es decir, se realiza una estimación de la duración y etapas del proyecto, así como de los costes tanto materiales como humanos que implicaría el desarrollo del sistema. Esto se encuentra detallado en el presente apartado del documento.
- **Estado del arte:** Durante esta fase se analiza exhaustivamente el estado del arte en materia de subtitulado en tiempo real en televisión. Se estudia el estado actual de los ASR, la normativa vigente en materia de subtitulado y los diferentes trabajos previos que pudieran resultar de utilidad para el desarrollo del presente proyecto.
- **Análisis:** Durante esta etapa se describen los requisitos funcionales del proyecto. Asimismo, se efectúa un exhaustivo estudio de todo lo que está relacionado con el mismo, y especialmente de los algoritmos de alineamiento de secuencias y de los reconocedores automáticos del habla.

- **Diseño:** Durante esta tarea se diseñan los algoritmos que se van a utilizar en el sistema, y también se especifican los elementos funcionales de los que va a constar mismo. Por otro lado, se determina el modelo de datos que se va a implementar en función de las necesidades existentes y las interfaces entre los elementos del sistema. Además se acota el funcionamiento que van a tener dichos elementos, por si fueran necesarias modificaciones futuras, ya que un sistema modular hace que sea muy sencillo actualizar o modificar los elementos.
- **Implementación:** En este período se implementan todas las funcionalidades propuestas en apartados anteriores. Se implementa desde el modelo de datos y los algoritmos utilizados hasta el entorno gráfico de la herramienta.
- **Pruebas:** La fase de pruebas engloba todas las comprobaciones efectuadas sobre todos y cada uno de los elementos y funcionalidades implementados durante la fase de codificación. Así, se evalúa el funcionamiento del sistema en diversos entornos, lo cual permite detectar los posibles errores y problemas no considerados previamente.
- **Despliegue:** Corresponde a la puesta en marcha del proyecto en un equipo, para poder ser utilizado si es necesario.
- **Redacción de la memoria del proyecto:** El último paso de la planificación comprende el período empleado en redactar la documentación relevante acerca del proyecto, como por ejemplo el diseño de la solución técnica o las pruebas efectuadas.

7. Conclusiones y trabajos futuros

7.1 Introducción

En el presente Trabajo de Fin de Grado se ha abordado una de las piezas clave del mayor problema existente en el subtitulado de programas en directo en televisión, un problema pendiente de resolver en todas las cadenas de televisión del mundo. El objetivo fundamental era el de desarrollar e implementar una serie de algoritmos que permitiesen, de una forma secuencial, aplicar la técnica de *word spotting* primero a través del alineamiento y ajustar los tiempos de referencia de las palabras que conforman los subtítulos después del mismo. Una vez generados esos algoritmos, su integración en cualquier tipo de sistema más complejo es inmediata. En torno a ese núcleo central, se ha generado un sistema capaz de realizar el proceso completo, esto es, aportar una solución al problema de la sincronización en subtítulos en los dos escenarios existentes en televisión. Las pruebas han sido exhaustivas para diferido, y para directo se ha tratado de emular las mismas condiciones, debido a la imposibilidad de trabajar en un entorno real durante el desarrollo del proyecto.

Debido a la existencia de varios escenarios, se planteó un sistema modular con interfaces entre elementos que facilitasen la comunicación entre ellos, así como la simplicidad y reutilización de código, e incluso la escalabilidad. Porque debido a esta arquitectura es muy sencillo extraer un módulo y sustituirlo por otro, o bien agregar nuevas formas de alineamiento o generación de subtítulos por ejemplo si se implementasen en un futuro.

A continuación se exponen las conclusiones extraídas y las líneas futuras de trabajo que pudieran resultar interesantes de cara a mejorar el trabajo desarrollado.

7.2 Conclusiones

Probablemente la parte más costosa fue la de generar y adaptar los algoritmos centrales sobre los que gira el sistema. Si bien es cierto que el de alineamiento se puede encontrar en pseudocódigo, también lo es que, como se explicó en la sección 4.3, los algoritmos existentes son a nivel de carácter. Hay que tener en cuenta la complejidad de extender esto a nivel de palabras, donde éstas coinciden, o no coinciden pero deben ir alineadas, o son muy parecidas pero no deben ir alineadas, etc. Fue necesario realizar un gran esfuerzo en ese sentido para lograr estimar unos umbrales de decisión que funcionasen en prácticamente todos los escenarios, y esta información fue obtenida mediante aproximaciones sucesivas a base de pruebas y modificaciones, dada la imposibilidad de obtenerla de otro modo. Por otro lado, el algoritmo de inferencia de tiempos también es complejo, ya que para poder observar los resultados sucede algo similar: no hay mejor forma de probar que el visionado directo. Es un modo muy trabajoso de efectuar las pruebas, pero una vez superados estos obstáculos y con una base algorítmica firme, se pudo derivar esfuerzo a implementar el resto del sistema.

Habiendo cumplido el requisito principal, que es el de haber desarrollado un método para alinear y calcular tiempos de referencia, el siguiente trabajo fue el de diseñar e implementar cada uno de los módulos de la herramienta. De este modo era posible cubrir la totalidad de casos posibles, teniendo siempre presente el uso que se podría dar en escenarios de subtitulado en televisión a ésta. Se trabajó en un diseño concurrente y escalable, que gestionase los recursos de la máquina donde se ejecutara convenientemente y que generase subtítulos con facilidad (o al menos una primera versión de los mismos fácilmente editable, para diferido por ejemplo).

Se puede concluir que, de acuerdo con las pruebas realizadas, que los algoritmos diseñaos e implementados funcionan razonablemente bien en los escenarios cubiertos por el proyecto, por lo que es de esperar que funcionen en otros entornos con una acústica más complicada, entre otras cosas porque el algoritmo de inferencia de tiempos desarrollado prevé que los ASR son imperfectos y permite obtener buenos resultados aun si hay una tasa de errores acotada de alineamiento. Lógicamente, debido a que en los casos de directo el rehablador genera subtítulos con un retardo determinado, es necesario algún mecanismo en el receptor TDT de los clientes que permita retrasar el vídeo ese retardo, de modo que los subtítulos se los ofrezca perfectamente sincronizados. Si para un instante dado de video su subtítulo el rehablador lo proporciona 15 segundos más tarde, es evidente que no se puede ofrecer en emisión los subtítulos sin ese retardo, porque el vídeo ya se emitió. Lo que ofrece el presente sistema es que corrige de manera individual todos los retardos variables de los subtítulos, por lo que si en el emisor o el receptor se contiene el vídeo durante un pequeño offset de tiempo, en la televisión del usuario se podría ver todo en sincronía, de modo que se solventase este grave problema que sufre un amplio sector de la sociedad.

Demostrado que el sistema completo de sincronización con rehablado es algo factible, se puede concluir que, en un futuro inmediato, la integración del sistema en las cabeceras de emisión de las televisiones es un objetivo abordable.

7.3 Trabajos futuros

En este caso concreto, hay múltiples vías de mejora del trabajo desarrollado. Esto se debe, en general, a que todas las operaciones, algoritmos y procesos que suceden durante la ejecución del programa tienen una componente de aleatoriedad importante, por lo que la inserción de las mejoras oportunas puede permitir incrementar las prestaciones del sistema.

La primera y más evidente posible mejora es la de ajustar y afinar los algoritmos que se ejecutan durante el proceso completo de ejecución del sistema. Tras realizar numerosas pruebas se estimaron unos porcentajes mínimos de parecido entre palabras y secuencias de palabras que, si se superaban, determinaban que dichas palabras o secuencias estaban alineadas. Esos porcentajes funcionan razonablemente bien en todos los escenarios considerados, pero es necesario ajustarlos más finamente, de modo que se maximicen así las prestaciones del algoritmo de alineamiento. En cuanto a la inferencia de tiempos,

el algoritmo actualmente reparte una duración entre un número determinado de letras, por igual. Esto es una aproximación razonable cuando se reparte la duración entre pocas palabras (por ejemplo, repartir un segundo entre dos palabras, una de dos letras y una de ocho, implica que a la primera palabra le corresponden 0,2 segundos y a la segunda 0,8). Además, esto sería preciso si el locutor hablase a ritmo constante, pero esto no es así. Hay cambios de ritmo que se producen a priori aleatoriamente que podrían provocar un peor resultado en el cálculo de tiempos. Por ello, una mejora significativa sería mejorar este algoritmo y hacerlo adaptativo, de manera que fuera capaz de ajustarse a estos cambios de ritmo y generar un mejor resultado. Por último, podría aparecer en un futuro un algoritmo de alineamiento más potente que el utilizado. Es, por tanto, trabajo futuro el investigar en ese ámbito e implementar, si fuese necesario, nuevos algoritmos mejorados o con mejores características que el de Needleman-Wunsch, además de automatizar la obtención de los umbrales de decisión.

Otra mejora que incrementaría las garantías del sistema es la de introducir un módulo detector de actividad vocal a la hora de transcribir el audio original. Actualmente esa entrada se pasa directamente por el ASR, que si de por sí produce muchos errores, con las distorsiones procedentes del audio no vocal este problema se agrava. Utilizando un detector de actividad vocal se restringiría el audio recibido a voz, mejorando la capacidad del módulo de reconocimiento de voz para generar una transcripción adecuada y, por tanto, proporcionando un texto en el que encontrar más coincidencias al alinear, aumentando las prestaciones del sistema completo.

Para disponer de una potente herramienta de generación de subtítulos, otro elemento fundamental es el Parser. Dado que no suponía un factor determinante para el proyecto, se implementó un generador de subtítulos relativamente sencillo que crea los subtítulos cumpliendo las normas básicas de tamaño de líneas y particionado. No obstante se podría trabajar en este sentido para crear un componente más potente de generación de subtítulos, que además de cumplir esas especificaciones permita editar subtítulos o proporcione otras opciones interesantes como escoger la posición, el formato, etc.

Para concluir, es preciso realizar pruebas sobre entornos no considerados en este proyecto para determinar su viabilidad, tales como programas deportivos o con una acústica completamente diferente, comprobar cómo se comportaría el sistema ante determinados eventos como las desconexiones en directo de los informativos, etc.

8. Repercusiones del proyecto desarrollado

8.1 Introducción

El proyecto desarrollado, como se ha visto a lo largo del documento, es de una importancia vital debido a la naturaleza del problema que pretende resolver. No existe actualmente nada que permita emitir subtítulos sincronizados en directo, por lo que cualquier mejora en este sentido es más que bienvenida por todo el conjunto de los ciudadanos. Por eso, este trabajo posee ciertas expectativas de futuro. Se algunas de las pruebas realizadas se han presentado como demostraciones para diversas entidades, de modo que se pueda dar salida a esta utilidad. En caso de que se implantara en un entorno real supondría un gran éxito, debido precisamente a que es algo completamente novedoso. Se trata de una técnica de sincronización automática útil en muchas situaciones, lo cual resulta atractivo desde un punto de vista empresarial y además es necesario tener en cuenta el factor de sensibilización social. Se tiende a proteger a los más desfavorecidos, por lo que cualquier empresa o entidad que lograse realizar un avance de estas características hacia la accesibilidad de la información obtendría un magnífico reconocimiento por parte de la sociedad.

8.2 Repercusiones

A corto – medio plazo se puede distinguir dos grandes propuestas realizadas para utilizar la herramienta desarrollada en un entorno real. Ambas propuestas se encuentran en fase de estudio en este momento, pero el hecho de que dos entidades tan importantes como RTVE y Telefónica estén considerando esas propuestas es esperanzador. A continuación se describen someramente cada una de ellas.

8.2.1 Colaboración con RTVE

En base a los resultados obtenidos en durante el desarrollo del proyecto, se ha definido una propuesta para integrar el sistema desarrollado en la plataforma de emisión de RTVE. De este modo, se podría ofrecer la opción de televisión con subtítulos sincronizados para los usuarios a través del portal web de televisión de RTVE. De esta manera, existiría una opción de televisión quasi-directo con subtítulos sincronizados.

Este es un gran paso de cara a la implantación de un posible sistema de sincronización a nivel de emisión televisiva. Si la propuesta llegase a término y los resultados en la web de RTVE a la carta, donde se pueden ver emisiones en directo, fueran satisfactorios, se podría continuar investigando y mejorando en este sentido.

8.2.2 Colaboración con Telefónica y RTVE

Por el mismo motivo, esto es, gracias a los buenos resultados obtenidos en cuanto a la sincronización, también se ha definido una propuesta para integrar el sistema en la plataforma Imagenio de Telefónica. La idea es análoga a la de RTVE, ofrecer la posibilidad de televisión con subtítulos sincronizados esta vez a los usuarios de Imagenio, a través de un canal IPTV adicional. Del mismo modo que la web de RTVE, se ofrecería un canal de quasi-directo con subtítulos sincronizados.

El objetivo del proyecto sería, en caso de ser aceptado, la realización de una prueba piloto para la creación de un canal en IPTV en el que se emita uno de los canales de RTVE con la sincronización. Este canal se emitiría ligeramente retardado respecto al original de RTVE unos 20 segundos aproximadamente, tiempo probablemente suficiente para corregir todos los retardos variables aparecidos en la emisión de los subtítulos originales. El sistema se probaría con diferentes programas de televisión que actualmente se subtitulan en directo, como por ejemplo los informativos de RTVE, 59 segundos, Las mañanas de la 1, programas de Teledeporte, etc.

8.3 Conclusiones

El hecho de que el proyecto resulte importante y que tenga unas expectativas de implantación de esta índole, aun estando en una fase de desarrollo temprana si se considera para ser implantado en un entorno real (ya que, lógicamente, no es trivial ni inmediato insertar el trabajo en una arquitectura ya implementada y operativa como puede ser por ejemplo la de RTVE) es algo muy significativo. Implica que se ha desarrollado una solución a una problemática pendiente en todas las televisiones del mundo y que, por tanto, es algo útil tanto para las personas como para la investigación. Para las primeras porque podría mejorar su calidad de vida. Para la investigación porque permitiría sentar unas bases en este ámbito, así como probablemente instaría a otros investigadores a buscar otras soluciones a este tipo de situaciones, siempre con vistas a mejorar la calidad de vida de todo aquel que lo necesite.

9. Referencias

[1] AENOR. Subtitulado para personas sordas y personas con discapacidad auditiva. Subtitulado a través del tetelexto. UNE 153101. Madrid: AENOR, 2003.

[2] España. Ley 7/2010, de 31 de marzo, General de Comunicación Audiovisual. Boletín Oficial del Estado, 1 de abril de 2010, núm. 79, p. 30157.

[3] Centro Español de Subtitulado y Audiodescripción. Seguimiento de los servicios de accesibilidad en la TDT 2011.

[4] Centro Español de Subtitulado y Audiodescripción. Informe de seguimiento de accesibilidad en la TDT: enero-febrero 2012.

[5] M. de Castro, M. de Pedro, B. Ruiz and J. Jimenez: "PROCEDIMIENTO Y DISPOSITIVO PARA SINCRONIZAR SUBTÍTULOS CON AUDIO EN SUBTITULACIÓN EN DIRECTO", Patent Id. P201030758, Oficina Española de Patentes y Marcas, Mayo/2010.

[6] M. de Castro, D. Carrero, L. Puente and B. Ruiz: "Real-time subtitle synchronization in live television programs," Broadband Multimedia Systems and Broadcasting (BMSB), 2011 IEEE International Symposium on , vol., no., pp.1-6, 8-10 June 2011 doi: 10.1109/BMSB.2011.5954889

[7] Mercedes de Castro, Luis Puente, Julián Hernández (2011). Synchronized Subtitles in Live Television Programs, 4th International Media for All Conference, London, UK, July 2011.

[8] J.E. Garcia, A. Ortega, E. Lleida, T. Lozano, E. Bernues, D. Sanchez, "Audio and text synchronization for TV news subtitling based on Automatic Speech Recognition," IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, BMSB '09, vol., no., pp.1-6, 13-15, May 2009.

[9] Romero-Fresco, Pablo (2011) Subtitling through Speech Recognition: Respeaking, Manchester: St Jerome.

[10] Needleman, S.B. and Wunsch, C. D. (1970): "A general method applicable to the search for similarities in the amino acid sequence of two proteins". *Journal of molecular biology* (Elsevier) 48 (3): pp. 443-453.

[11] Smith TF, Waterman MS (1981): "Identification of common molecular subsequences." *J Mol Biol.* 147 (1): pp. 195-7